



Précision p-adique

Tristan Vaccon

► To cite this version:

Tristan Vaccon. Précision p-adique. Mathématiques générales [math.GM]. Université de Rennes, 2015. Français. NNT : 2015REN1S032 . tel-01205269v2

HAL Id: tel-01205269

<https://theses.hal.science/tel-01205269v2>

Submitted on 8 Oct 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



THÈSE / UNIVERSITÉ DE RENNES 1
sous le sceau de l'Université Européenne de Bretagne

pour le grade de
DOCTEUR DE L'UNIVERSITÉ DE RENNES 1

Mention : Mathématiques et applications

Ecole doctorale Matisse

présentée par

Tristan Vaccon

préparée à l'UMR 6625 CNRS-IRMAR
Institut de Recherche Mathématique de Rennes
U.F.R. Mathématiques

Précision p -adique

*Applications en calcul formel,
théorie des nombres et cryptographie*

Thèse soutenue à Rennes
le 3 juillet 2015

devant le jury composé de :

Jean-Charles FAUGÈRE

DR INRIA, CRI Paris-Rocquencourt / rapporteur

Kiran S. KEDLAYA

Professeur, University of California San Diego /
rapporteur

Jean-Marc COUVEIGNES

Professeur, Université de Bordeaux / examinateur

Grégoire LECERF

CR CNRS, École Polytechnique / examinateur

Marie-Françoise ROY

Professeure émérite, Université de Rennes 1 /
examinatrice

Xavier CARUSO

CR CNRS, Université de Rennes 1 / directeur de thèse

Résumé

Les calculs effectifs sur les p -adiques ont de nos jours de nombreuses applications. L'une des plus célèbre est leur usage pour divers algorithmes de comptage de points sur des courbes définies sur des corps finis. Cependant, par essence, il n'est en général pas possible de manipuler numériquement les nombres p -adiques autrement qu'à précision finie. En pratique, cela revient à travailler avec des quantités de la forme $a + O(p^n)$ (et éventuellement dans \mathbb{Q}_p^d , des quantités de cette forme sur chaque coordonnée). La précision p -adique s'intéresse au suivi de la précision de ces approximations.

Nous proposons un nouveau modèle pour étudier et suivre la précision, qui consiste, plutôt qu'à regarder des $O(p^n)$ sur chaque coordonnée, à considérer des \mathbb{Z}_p -réseaux dans \mathbb{Q}_p^d . Nous avons alors le résultat suivant : étant donnée une approximation $x + H$ d'un élément $x \in \mathbb{Q}_p^d$ par un réseau H , assez petit, la précision sur $f(x + H)$ est essentiellement donnée par l'approximation de Taylor de f au premier ordre : $f(x + H) = f(x) + f'(x) \cdot H$. Ceci permet de ramener le suivi de la précision de manière qualitative à un simple calcul de différentielle. De plus, en connaissant les normes des dérivées d'ordre supérieur de f , nous pouvons préciser quantitativement quand le réseau H est assez petit.

Afin de rendre ce résultat effectif, nous fournissons une méthode, dite méthode adaptative, pour atteindre le comportement de la précision donnée par l'approximation au premier ordre. Nous illustrons ces méthodes sur divers exemples tirés de l'algèbre linéaire ou des équations différentielles p -adiques.

Une autre question naturelle autour de \mathbb{Q}_p , et des systèmes polynomiaux qui peuvent être définis sur \mathbb{Q}_p , est celle de savoir quelles bases de Gröbner peuvent être calculées. Nous proposons une réponse en trois temps.

Dans un premier temps, nous donnons une condition, Zariski ouverte, sur les systèmes polynomiaux (non surdéterminés) pour qu'une base de Gröbner puisse être calculée, grâce à une adaptation de l'algorithme F5-Matriciel. Ceci nous permet d'obtenir la différentiabilité du calcul d'une base de Gröbner sur les ouverts correspondant et nous estimons son module de continuité. Cette condition ouverte est dense pour grevlex modulo la conjecture de Moreno-Socias.

Dans un second temps, nous nous intéressons au changement d'ordre monomial. Grâce à l'algorithme FGLM, légèrement adapté, il est possible de passer, pour des idéaux de dimension zéro, d'une approximation d'une base de Gröbner pour un ordre monomial à une autre pour un second ordre monomial. La perte de précision est donnée par les facteurs invariants de la matrice de changement de base dans l'anneau de coordonnées correspondant.

Enfin, dans un troisième temps, nous ajoutons une approche tropicale aux réponses précédentes. Il s'agit de travailler avec une définition de base de Gröbner tropicales issue de la géométrie tropicale et prenant en compte la valuation. Nous montrons que de telles bases peuvent être calculées par une version adaptée de l'algorithme F5-Matriciel. De plus, à précision finie, elles sont calculables grâce à cet algorithme sur un ouvert plus gros et toujours dense des systèmes polynomiaux avec degrés bornés et non surdéterminés. Pour des idéaux de dimension zéro, il est possible d'utiliser un algorithme FGLM pour passer d'une base de Gröbner tropicale à une base de Gröbner classique. Ceci permet, en dimension zéro, de donner une dernière réponse au problème du calcul de base de Gröbner à précision finie : si $F = (f_1, \dots, f_n)$ est une suite régulière de polynômes homogènes de $\mathbb{Q}_p[X_1, \dots, X_n]$ connue à une précision suffisante, alors on peut calculer une base de Gröbner tropicale de $\langle F \rangle$, puis en déduire une base de Gröbner classique de $\langle F \rangle$. Le caractère suffisant est donné par des mineurs de matrices de Macaulay et des facteurs invariants de matrices de changements de base. Ce résultat s'adapte naturellement au cas où $F = (f_1, \dots, f_n)$ est telle que ses composantes homogènes de plus haut degré (f_1^h, \dots, f_n^h) forment une suite régulière.

Remerciements

Je tiens tout d'abord à remercier mon directeur de thèse Xavier Caruso, sans qui cette thèse n'existerait pas. Je lui suis éternellement reconnaissant pour sa disponibilité, ses conseils toujours avisés et pour m'avoir guidé tout au long de cette thèse sur les chemins de la recherche. J'espère qu'il saura me pardonner de ne pas avoir accompli le dixième de ce qu'il pouvait me suggérer de faire.

Je remercie vivement Kiran Kedlaya et Jean-Charles Faugère de m'avoir fait l'honneur de rapporter ce manuscrit. Leurs commentaires précis et leurs remarques auront, j'en suis sûr, permis une amélioration significative de la qualité de ce dernier. Je suis aussi très reconnaissant envers Jean-Marc Couveigne, Grégoire Lecerf, et Marie-Françoise Roy d'avoir accepté d'être membres du jury de cette thèse. Tous me font un grand honneur en s'intéressant ainsi à mon travail.

Tout au long de cette marche vers la recherche, j'ai eu la chance de faire des rencontres mathématiques qui ont été pour moi décisives. Par ordre chronologique, je remercie Guénaél Renault pour avoir accepté d'encadrer mon stage de fin de L3 et pour avoir continué à suivre mes progrès depuis, jusqu'à finalement me donner son intuition décisive sur l'étude de FGLM et que nous devenions co-auteurs. Je remercie Kazuhiro Yokoyama pour avoir accepté d'encadrer mon stage de fin de M1. Sa rigueur et son sérieux dans mon encadrement ne furent dépassés que par sa générosité. Je garde un souvenir inoubliable de cette période de l'été 2010, et je n'ai de cesse, depuis, que de tenter de retrouver les conditions pour la reproduire. Sans ces deux rencontres, je ne sais vraiment pas quel genre de mathématiques auraient eu mes faveurs. Je remercie David Lubicz pour avoir veillé sur moi, en particulier lors des débuts de ma thèse. Je n'oublierai jamais le fait qu'il ait cru en moi et en l'étude des bases de Gröbner sur \mathbb{Q}_p , ni le fait que par son influence, il ait rendu financièrement possible mes exposés à nombres de conférences, ainsi que permis de faire venir à Rennes ceux que je voulais voir exposer.¹

D'autres rencontres auront été décisives pour moi au cours de cette thèse. Je remercie en particulier mon co-auteur David Roe, sans qui il n'est pas clair qu'une recherche sur la précision p -adique aurait même pu avoir lieu ! Je remercie aussi mon co-auteur Pierre Lairez. À quoi tiennent les rencontres ? Quelques semaines avant les JNCF 2013, je ne savais même pas qu'elles existaient, ni même pendant un moment, l'envie d'y participer et pourtant, au hasard des placements dans les chambres du CIRM, je n'oublierai pas le moment où j'ai vu à l'œuvre la magie des mathématiques avec ce rendez-vous inattendu entre la question que tu portais depuis déjà quelques temps et la promesse de réponse que porte l'étude différentielle de la précision. Je n'hésiterais pas à placer ce séjour au CIRM et cette rencontre comme l'un des points de basculement majeurs dans ma thèse : après lui, la route à prendre était devenue très claire !

Tout au long de cette thèse, j'aurais été un visiteur régulier de l'équipe PolSys du LIP6, et je les remercie pour leur accueil chaleureux, en particulier (outre ceux que j'aurais déjà cités) Alexandre, Ivan, Jérémy, Jules, Louise, Pierre-Jean et Thibaut. Je remercie aussi Daniel Lazard pour ses conseils et une longue discussion sur la conjecture de Moreno-Socias. Cela aura toujours été un plaisir pour moi, après chaque long trajet entre Argenteuil et Jussieu, d'atteindre l'étage de l'équipe PolSys !

Peut-être celui avec qui j'ai le plus de conférences en commun, je remercie Luca De Feo, qu'il m'est toujours très agréable de retrouver pour parler de courbes elliptiques et d'équations différentielles. Aussi rencontré en conférence, je remercie chaleureusement Sam Derbyshire, avec qui il est toujours délicieux de parler de cohomologies, et Bruno Winckler, fameux chercheur en pokémathématiques.

J'ai eu la chance de pouvoir me former en donnant des exposés à de nombreux séminaires et conférences, et je remercie en particulier (dans le désordre) Alin Bostan pour l'équipe SpecFun de l'INRIA Saclay, Elias Tsigaridas pour l'équipe PolSys, Delphine Boucher pour les JNCF et le séminaire de Calcul Formel de Rennes, Marie-Françoise Roy pour MAP, Alan Hertgen et Jean-Baptiste Boyer pour les rencontres du troisième cycle, Christophe Dupont pour le séminaire de géométrie de Rennes, Jérémy Le Borgne, Sandrine Caruso et Arnaud Girand pour le séminaire Pampers des doctorants en géométrie de Rennes, Hélène Hivert pour le séminaire Landau des doctorants en analyse de Rennes et Thibaut Dehevels pour le groupe de travail de l'ENS Rennes.

1. Je n'oublierai pas non plus que, grâce à son rire et son enthousiasme si communicatif, il soit la personne capable de me faire rire aux larmes le plus facilement.

Je remercie aussi les organisateurs des différents groupes de travail qui m'ont permis d'apprendre beaucoup et d'exposer parfois, Lionel Fourquaux, Matthieu Romagny, et Yvan Ziegler, et dans le même temps, je remercie tous les participants aux deux groupes que j'ai pu organiser et sans qui je ne saurais probablement rien sur les variétés de Shimura et les groupes de Lie.

Une condition nécessaire (mais, hélas, certainement pas suffisante) pour pouvoir faire une recherche mathématique de qualité raisonnable est d'avoir des conditions matérielles la rendant agréable. Je suis heureux d'avoir pu effectuer ma thèse à l'IRMAR où cette condition a été remplie. Je remercie en particulier les directeurs de l'équipe de Géométrie Algébrique, Antoine Chambert-Loir puis Julien Sebag, pour avoir accepté toutes mes demandes de financements. Cela aura été une grande fierté pour moi d'appartenir quelques années à cette équipe et d'y côtoyer des chercheurs aussi remarquables, que je remercie tous. Je remercie en particulier Michel Gros, avec qui il a toujours été un plaisir de participer à un même jury d'oraux blancs ou de parler du Japon. Je remercie aussi les organisateurs du séminaire de géométrie algébrique de m'avoir permis d'inviter quelques orateurs.

Un des avantages non négligeables de travailler à l'IRMAR, c'est de pouvoir compter sur une équipe administrative à l'efficacité redoutable, et je remercie en particulier Chantal Halet, Hélène Rousseaux et Marie-Aude Verger, ainsi que Morgane Leray, Nelly Loton, Elise Ramos, Marie-Annick Paulmier, Olivier Garo, Patrick Pérez, Maryse Collin, Dominique Hervé, Marie-Annick Guillemer, Élodie Cottrel, Anne-Joëlle Chauvin et leurs collègues. Je remercie aussi Bachir Bekka pour sa très efficiente direction de l'IRMAR.

Au sixième étage de l'IRMAR, j'ai aussi eu la chance de pouvoir côtoyer "l'équipe d'à côté," avec qui il aura toujours été un plaisir d'interagir. Je remercie en particulier (et à nouveau) Delphine Boucher et David Lubicz pour m'avoir permis d'inviter nombre d'orateurs à leurs séminaires respectifs : l'avancement de ma thèse aurait certainement été beaucoup plus lent sans eux.

Une autre étape importante lors de ma thèse aura été l'organisation des Rencontres Doctorales Lebesgue 2014 et pour cela, je remercie le Labex Lebesgue (ainsi que nos autres soutiens), nos orateurs, les autres co-organisateurs et en particulier Arnaud Girand, sans qui rien n'aurait été possible.

Parmi les activités importantes d'un doctorant, l'enseignement en est une qui compte. J'ai eu la grande chance de pouvoir faire le mien à l'ENS Rennes. Je remercie le département de mathématiques pour son accueil, en particulier Benoit Cadre, Arnaud Debussche, Thibaut Deheuvels, Michel Pierre et Élodie Le Quoc, ainsi que les doctorants de l'ENS Rennes, Charles-Édouard, Maxime, Marie, Sylvain, Mac et Quentin. Je remercie aussi les professeurs du cours dont j'ai donné le TD, Yongquan Hu, Xavier Caruso et Julien Sebag. J'ai par ailleurs pu, avant de faire ma mission d'enseignement, donner des colles aux lycées Joliot-Curie et Chateaubriand, et je remercie pour cela les professeurs Gabillard et Hartmann de m'avoir fait l'honneur de compter sur moi pour coller.

Si la vie à l'IRMAR était si agréable, c'est en grande partie grâce à mes collègues doctorants. Je remercie donc, sans compter les multiplicités, Alexandre, Arnaud, Axel, Blandine, Camille, Charles, Christian, Damien, Élise, Fabien, Federico, Felipe, Florian, Hélène, Jean-Philippe, Julie, Kodjo, Loubna, Mohamed, Nestor, Ophélie, Pierre-Yves, Renan, Romain, Türkü, VMK, YZ, et tous les autres.

Je tiens aussi à remercier mes collègues de bureau (et du bureau mitoyen) grâce à qui venir à l'Université a toujours été un plaisir : Basile, Cécile, Gabriel², Gwezheneg, Jérémy (mon *senpai*), Manh Tu, MouTon, et Sachio.

Si j'ai ainsi pu effectuer ma thèse, c'est grâce à l'excellente formation que j'ai pu recevoir à l'ENS Rennes et à l'Université de Rennes 1. Je remercie particulièrement tous mes professeurs au cours de cette période. Et je n'aurai certainement jamais pu venir à Rennes sans l'inégalable qualité des cours de M. Bertrand et M. Pépin, mes professeurs de mathématiques au lycée Condorcet à Paris, et avant eux, M. Eugénie, M. Tamby et Mme Guégan³, mes professeurs de mathématiques au lycée Jean Jaurès à Argenteuil.⁴

Venir à Rennes aura été une chance incroyable pour moi, non seulement pour la qualité de la formation que j'ai pu y recevoir, mais également pour la qualité de l'ambiance de travail (mais pas seulement) qui y règne. Faire partie de la promotion Maths 2008 restera un moment marquant dans ma vie, et j'en remercie tous les membres. Il serait injuste de n'en citer qu'une partie. Je pense

2. Gabriel que je remercie par ailleurs d'avoir relu les remerciements de ce tapuscrit.

3. Que ceux-ci me pardonnent si jamais je me suis trompé sur l'orthographe de leurs noms.

4. Et je n'oublie pas non plus mes professeurs au collège à Argenteuil !

aussi à certains que j'ai pu rencontrer qui venaient d'autres promotions et d'autres départements (en particulier Laurent, Amélie, et Arthur).

Je suis heureux de pouvoir encore revoir régulièrement mes camarades de prépa qui ont décidé de se lancer aussi dans les mathématiques, et tout au long de cette thèse, cela aura toujours été un plaisir de pouvoir régulièrement parler maths avec Alexandre et Julien, en particulier lors de mes rares passages à Paris.⁵ En parlant de Paris, je ne peux pas ne pas remercier spécialement TP, peut-être celui avec qui j'ai le plus écumé les restaurants de la rue Saint-Anne !

Comme il n'y a pas que la thèse dans la thèse, je remercie mes anciens partenaires du foot du vendredi soir au bas de l'IRMAR, du tennis du mardi soir (tout particulièrement Benoit et Axel, qui sont certainement ceux avec qui j'aurais le plus tapé dans la balle) et du golf du mardi midi (en particulier Camille, Nicolas, Pierre et Benoit). Je remercie aussi mes camarades lors de mes cours de japonais, en particulier ceux que j'ai côtoyés le plus longtemps, Charly, Vincent, Youssef, Romain et bien sûr Siargey, et je remercie aussi ma professeure de japonais, Maliko Oka que je suis désolé d'avoir embêté en restant pendant tant d'années au cours des élèves de deuxième-troisième année !

Il va sans dire que sans eux rien n'aurait eu lieu, je ne trouve pas de mots assez forts pour remercier assez fortement mes parents. Je n'arriverai certainement jamais assez à les remercier pour leur soutien et leur confiance, et pour avoir fait de moi ce que je suis maintenant. Je pense aussi à ma grand-mère, cette thèse ayant eu le prix de ne plus me permettre que de la voir rarement, ce qui restera certainement un des seuls regrets pour moi dans celle-ci. Je n'oublie pas non plus ma famille dans le sud-est, et en particulier mes cousins Alex, Aline et Arnaud, qui auront toujours su m'accueillir chaleureusement (que le CIRM est bien placé !).

Je remercie bien sûr aussi Denis, Evelyne, Léa et Barbara pour leur accueil et leur soutien⁶ enthousiaste.

Enfin, il ne m'est pas possible de finir sans remercier celle qui aura donné des limites à toutes mes suites de Cauchy (pour une norme ultramétrique, il va sans dire). Salomé, même si tu n'as écrit aucune ligne de code ni le moindre bout d'aucune des preuves de cette thèse, celle-ci te doit beaucoup plus que tu ne pourrais le croire, et je ne saurais te remercier assez pour avoir été à mes côtés tout au long de son déroulement.

5. Je remercie aussi Anaïs, qui me doit un tennis.

6. En particulier culinaire !

Table des matières

| | |
|---|---------------|
| Introduction | 15 |
| Contexte | 15 |
| Un peu d'histoire | 17 |
| Nombres p -adiques | 17 |
| Algorithmique et p -adiques | 17 |
| Passage par les p -adiques | 18 |
| Algorithmes p -adiques par nature | 19 |
| Théorie des bases de Gröbner | 19 |
| Géométrie tropicale | 19 |
| Contributions | 20 |
| Précision p -adique | 20 |
| Systèmes polynomiaux | 23 |
| Publications | 26 |
| Organisation du manuscrit | 27 |
| I. Précision différentielle | 29 |
| Résumé | 31 |
| Notations | 31 |
| Modèle de complexité | 32 |
| Contexte | 32 |
| Nombres p -adiques, algorithmique et précision | 32 |
| Représentation classique et méthode directe de suivi de la précision | 33 |
| Implémentation en pratique des p -adiques | 33 |
| Algorithmique détendue et en-ligne | 34 |
| 1. Méthode directe, applications et limites | 35 |
| 1.1. Suivi direct de la précision | 35 |
| 1.1.1. Précision sur un CDVF à précision finie | 35 |
| 1.1.2. Comparaison qualitative avec les réels | 36 |
| 1.2. Application de cette méthode : le cas de l'échelonnement en lignes | 38 |
| 1.2.1. Algorithme d'échelonnement en lignes par élimination gaussienne | 38 |
| 1.2.2. Comment pivoter | 39 |
| 1.2.3. Calcul de la forme échelonnée | 39 |
| 1.2.4. Un résultat plus précis | 40 |
| 1.3. Résolution de systèmes linéaires et forme normale de Smith | 40 |
| 1.3.1. Forme normale de Smith | 40 |
| 1.3.2. Calcul de la forme normale de Smith | 41 |
| 1.3.3. Résolution de systèmes linéaires. | 45 |
| 1.3.4. Application aux matrices de Hilbert | 45 |
| 1.4. Les limites des méthodes directes | 46 |
| 1.4.1. Un exemple naïf | 47 |
| 1.4.2. D'autres exemples | 47 |
| 2. Le lemme de précision | 51 |
| 2.1. Les réseaux comme modèle de précision | 51 |
| 2.1.1. Réseaux | 51 |
| 2.1.2. Séparer précision et approximation | 52 |
| 2.1.3. Types de précision : quelques réseaux particuliers | 53 |

| | | |
|-----------|---|-----------|
| 2.1.4. | Diffusion de la précision | 54 |
| 2.2. | Lemme principal : réseaux du premier ordre et précision | 54 |
| 2.2.1. | Applications différentiables | 54 |
| 2.2.2. | Images de réseaux par des applications différentiables | 55 |
| 2.3. | Caractérisation analytique des réseaux du premier ordre | 56 |
| 2.3.1. | Le cas des fonctions localement analytiques | 56 |
| 2.3.2. | Caractérisation par une équation différentielle | 60 |
| 2.3.3. | Démonstration de la Proposition 2.3.15 | 61 |
| 2.4. | Premières idées de mise en œuvre en pratique | 63 |
| 2.4.1. | Calcul en un passage | 63 |
| 2.4.2. | Calcul en deux passages | 64 |
| 2.4.3. | Remarques sur la surjectivité | 65 |
| 2.5. | Généralisation aux variétés | 65 |
| 2.5.1. | K -variétés différentiables | 65 |
| 2.5.2. | Données de précision | 66 |
| 2.5.3. | Généralisation du lemme principal | 67 |
| 2.5.4. | Un premier exemple | 68 |
| 3. | Applications du lemme, calcul de différentielles | 69 |
| 3.1. | Polynômes | 69 |
| 3.2. | Matrices | 72 |
| 3.2.1. | Multiplication | 72 |
| 3.2.2. | Déterminant | 74 |
| 3.2.3. | Polynôme caractéristique | 75 |
| 3.2.4. | Décomposition LU | 76 |
| 3.3. | Espaces vectoriels | 77 |
| 3.3.1. | Géométrie des grassmanniennes | 77 |
| 3.3.2. | Calcul différentiel | 78 |
| 3.3.3. | Implémentations et expériences | 80 |
| 4. | Précision en pratique : méthodes adaptatives | 83 |
| 4.1. | Échecs de certains calculs directs | 83 |
| 4.1.1. | Retour sur la suite de SOMOS-4 | 83 |
| 4.1.2. | Retour sur les arrangements | 84 |
| 4.2. | Une méthode adaptative | 84 |
| 4.2.1. | Illustration du problème | 84 |
| 4.2.2. | Illustration d'une solution | 85 |
| 4.3. | Applications | 86 |
| 4.3.1. | Conclusion sur SOMOS-4 | 86 |
| 4.3.2. | Conclusion sur les arrangements | 88 |
| 5. | Un exemple complet : résolution d'équations différentielles | 91 |
| 5.1. | Présentation du problème | 91 |
| 5.1.1. | Introduction et résultats principaux | 91 |
| 5.1.2. | Applications | 92 |
| 5.2. | Étude théorique, réseaux du premier ordre | 93 |
| 5.2.1. | La différentielle de Y_n | 93 |
| 5.2.2. | Réseaux du premier ordre | 95 |
| 5.2.3. | Conclusion sur la précision | 97 |
| 5.3. | Application effective, atteindre la borne | 97 |
| 5.3.1. | L'itération de Newton naïve ne suffit pas | 97 |
| 5.3.2. | Stabilisation de la méthode de Newton-Hensel : une méthode adaptative | 98 |
| 5.3.3. | Complexité | 99 |
| 5.4. | Implémentation | 99 |

| | |
|--|------------|
| II. Systèmes polynomiaux | 101 |
| Résumé | 103 |
| Contexte | 103 |
| Méthodes p -adiques et bases de Gröbner | 103 |
| Bases de Gröbner flottantes | 104 |
| Résolution de systèmes polynomiaux p -adiques | 104 |
| Notations | 105 |
| Modèle de complexité | 105 |
| Synthèse des résultats de cette partie | 106 |
| 6. Algorithmes classiques pour le calcul de bases de Gröbner | 109 |
| 6.1. Bases de Gröbner et applications | 109 |
| 6.1.1. Que sont les bases de Gröbner ? | 109 |
| 6.1.2. Pourquoi calculer des bases de Gröbner ? | 111 |
| 6.2. Calcul de bases de Gröbner et algèbre linéaire | 113 |
| 6.2.1. L'algorithme F5-Matriciél | 113 |
| 6.2.2. L'algorithme FGLM | 123 |
| 7. Algorithme F5-Matriciél et stabilité | 135 |
| 7.1. Présentation des résultats de cette partie | 135 |
| 7.2. Algorithme F5-Matriciél et calcul de bases de Gröbner à précision finie | 136 |
| 7.2.1. Problèmes de précision | 136 |
| 7.2.2. L'Algorithme F5-Matriciél-Faible | 138 |
| 7.2.3. Précision <i>vs</i> complexité | 142 |
| 7.3. Topologie et optimalité | 143 |
| 7.3.1. Continuité et optimalité | 143 |
| 7.3.2. Différentiabilité | 143 |
| 7.4. Implémentation | 144 |
| 7.4.1. Calculs directs | 144 |
| 7.4.2. De la stabilité | 145 |
| 7.5. Méthode de remontée sous l'hypothèse H2 | 145 |
| 7.5.1. Remonter une base de Gröbner | 145 |
| 7.5.2. Remontée à des points satisfaisant H1 et H2 | 146 |
| 7.5.3. Application | 148 |
| 7.6. Le cas affine et base de Gröbner réduite | 148 |
| 7.6.1. Le cas affine | 148 |
| 7.6.2. Calcul de la base de Gröbner réduite | 149 |
| 8. Stabilité de FGLM | 151 |
| 8.1. Stabilité de l'algorithme direct | 151 |
| 8.1.1. Un algorithme stabilisé | 151 |
| 8.1.2. Correction, terminaison, précision et complexité | 153 |
| 8.2. Cas d'un idéal en position générale | 157 |
| 8.2.1. Présentation de l'algorithme | 157 |
| 8.2.2. Correction, terminaison et précision | 159 |
| 8.3. Implémentation | 160 |
| 9. Une approche tropicale | 163 |
| 9.1. Introduction et motivations tropicales | 163 |
| 9.1.1. Résultats principaux | 163 |
| 9.1.2. Motivations tropicales | 164 |
| 9.2. Algorithmes F5-Matriciels tropicaux | 166 |
| 9.2.1. Un premier algorithme F5-Matriciél tropical | 166 |
| 9.2.2. Le cas des CDVF à précision finie | 171 |
| 9.2.3. Implémentation | 174 |
| 9.2.4. Un algorithme F5-Matriciél plus rapide | 174 |

Table des matières

| | |
|--|------------|
| 9.3. Un algorithme FGLM tropical | 179 |
| 9.3.1. Calcul des matrices de multiplication | 179 |
| 9.3.2. Application aux algorithmes FGLM | 182 |
| 9.4. Méthode tropicale pour des calculs de bases de Gröbner classiques | 184 |
| 9.5. Implémentation | 186 |
| Perspectives et questions ouvertes | 189 |
| Bibliographie | 191 |

Liste des Algorithmes

| | | |
|---------|--|-----|
| 1.2.3. | Échelonnement en lignes gaussien | 38 |
| 1.3.5. | SNFApprochee : Calcul de forme normale de Smith approchée | 42 |
| 1.3.8. | SNFPrecisee : Passage d'une forme approchée à la forme normale de Smith | 43 |
| 3.2.2. | Exemple de produits | 73 |
| 3.3.2. | Opérations aléatoires sur des sous-espaces vectoriels | 80 |
| 4.3.1. | SOMOS(a, b, c, d, n, N) | 86 |
| 4.3.2. | Liste des arrangements, stabilisé | 88 |
| 5.3.2. | L'algorithme de Newton-Hensel stabilisé | 98 |
| 6.1.9. | Division d'un polynôme par une famille de polynômes | 110 |
| 6.2.6. | L'algorithme de Lazard | 114 |
| 6.2.12. | Un premier algorithme F5-Matriciel | 116 |
| 6.2.19. | L'algorithme d'échelonnement sans choix du pivot | 120 |
| 6.2.22. | Algorithme F5-Matriciel avec signatures | 121 |
| 6.2.30. | Calcul d'une forme normale par les matrices de multiplication | 124 |
| 6.2.34. | Calcul des matrices de multiplication | 126 |
| 6.2.36. | Algorithme FGLM simplifié | 127 |
| 6.2.38. | Algorithme FGLM effectif | 128 |
| 6.2.39. | Update, échelonnement partiel | 129 |
| 6.2.48. | Algorithme FGLM pour un idéal en position générale | 132 |
| 6.2.53. | Algorithme FGLM pour un idéal en position générale à partir de grevlex | 133 |
| 7.2.7. | L'algorithme F5-Matriciel-Faible | 138 |
| 7.5.2. | L'algorithme de Remontée-Faible | 146 |
| 8.1.1. | Algorithme FGLM stabilisé | 152 |
| 8.1.2. | Update, forme normale de Smith approchée itérée | 152 |
| 8.2.1. | Algorithme FGLM stabilisé pour un idéal en position générale | 158 |
| 8.2.2. | Algorithme FGLM stabilisé pour un idéal en position générale à partir de grevlex | 159 |
| 9.2.3. | L'algorithme d'échelonnement tropical | 167 |
| 9.2.8. | Un premier algorithme F5-Matriciel | 168 |
| 9.2.22. | L'algorithme LUP tropical | 176 |
| 9.2.25. | Algorithme F5-Matriciel tropical avec signatures | 177 |
| 9.3.5. | Calcul d'une forme échelonnée réduite tropicale | 180 |
| 9.3.8. | Calcul des matrices de multiplication dans le cas tropical | 181 |
| 9.3.10. | Algorithmes FGLM tropicaux | 182 |
| 9.3.12. | Algorithmes FGLM tropicaux numériques | 183 |

Introduction

“This book was written using 100 % recycled words.”

Terry Pratchett, *Wyrd Sisters*

"The truth will always find a way to make itself known. The only thing we can do is to fight with the knowledge we hold and everything we have. Erasing the paradoxes one by one... It's never easy... We claw and scratch for every inch. But we will always eventually reach that one single truth. This I promise you."

Miles Edgeworth, *Phoenix Wright : Ace Attorney : Justice For All*

Contexte

On définit classiquement le calcul formel comme le domaine des mathématiques qui s'intéresse aux calculs et algorithmes sur des objets mathématiques à travers des représentations finies et exactes de ceux-ci. Comme exemple de tels objets, nous pouvons citer les nombres entiers, qui peuvent être représentés par leur écriture en base 2. Étant donnés deux tels nombres, le calcul formel s'intéresse au calcul de leur somme, ou de leur produit, différence, ou encore division euclidienne du premier par le deuxième, avec le résultat écrit selon la représentation choisie.⁷ Autant que possible, on souhaite pouvoir tester l'égalité de deux objets qui peuvent être donnés par des représentations distinctes. Étant donné un algorithme portant sur de tels objets, en calcul formel, on s'intéresse à montrer qu'il termine en temps fini et qu'il renvoie bien la représentation de l'objet attendu. On peut aussi s'intéresser à la complexité de l'algorithme, en particulier en estimant le nombre d'opérations élémentaires effectuées, ce qui permet d'évaluer le temps de calcul de cet algorithme.

Hélas, il est des objets mathématiques sur lesquels on pourrait souhaiter faire des calculs mais qui ne peuvent admettre de représentations finies et exactes. Il n'est pas besoin de chercher très loin pour trouver des exemples de tels objets : les nombres réels sont de ceux-là. Ceux-ci possèdent tous une représentation unique en base b de la forme $\pm \sum_{i \leq l} a_i b^i$ avec les $a_i \in \llbracket 0, b-1 \rrbracket$ (et a_i ne stationne pas en $b-1$). En conséquence, pour des raisons de non-dénombrabilité, presque tout nombre réel est non calculable, au sens où il n'existe pas d'algorithme (ou de machine de Turing) permettant de donner ses chiffres dans son écriture en base b .⁸ Malgré cela, diverses stratégies sont développées pour pouvoir tout de même faire des calculs sur les réels. La plus classique est l'usage des nombres flottants, qui constitue l'un des fondements de ce qu'on appelle le calcul scientifique. Pour une précision fixée n , il s'agit de se restreindre à travailler avec des nombres de la forme $s \times m \times b^e$, où $s \in \{-1, +1\}$ est le signe, b est la base d'écriture, e est un entier, et m un entier naturel plus petit que b^n . Ces nombres approchent tous les réels, et le nombre de chiffres de m dans l'écriture précédente correspond à la précision de l'approximation effectuée. Néanmoins, divers

7. Pour citer d'autres objets d'intérêt classiques du calcul formel, on pourrait parler, par exemple, des nombres rationnels, des corps de nombres, des corps finis, mais aussi des polynômes, des fractions rationnelles (et de leurs intégrales), des matrices ... tous ces derniers étant pris sur un corps où l'on peut faire du calcul formel.

8. Cela ne dépend pas de b .

phénomènes, lorsqu'on atteint les bornes fixées sur la précision, ou dus aux arrondis font que les calculs sur les flottants sont à prendre avec précaution : le nombre flottant pris comme résultat de la somme de deux d'entre eux n'est pas forcément une approximation aussi bonne du réel qu'il cherche à approcher que ne le sont les deux nombres flottants pris comme termes. Par exemple, $1001 \times 2^3 + (-1000 \times 2^3)$ donne 1×2^3 , nombre sur lequel on ne dispose plus que d'un seul chiffre de précision. Ainsi, après une succession d'opérations élémentaires (somme, soustraction, multiplication, division), il est possible d'obtenir comme résultat des nombres flottants de la forme $s \times 1 \times b^e$ ou 0, autrement dit sans chiffre de précision. La difficulté lorsqu'on s'intéresse à des algorithmes sur ce type d'objet est alors d'en évaluer la stabilité : si on donne en entrée de l'algorithme des approximations des nombres réels, est-ce que la sortie de l'algorithme est bien une approximation raisonnable de ce que l'algorithme donnerait de manière formelle ?

Une autre stratégie, plus sophistiquée, est celle de l'arithmétique d'intervalles. Elle consiste à approcher un réel x par un couple de flottants $(x_{\text{inf}}, x_{\text{sup}})$, avec $x \in [x_{\text{inf}}, x_{\text{sup}}]$. Les opérations usuelles s'écrivent alors comme des opérations sur les couples de flottants. Par exemple, l'addition s'écrit $(x_{\text{inf}}, x_{\text{sup}}) + (y_{\text{inf}}, y_{\text{sup}}) := (x_{\text{inf}} +_{\text{inf}} y_{\text{inf}}, x_{\text{sup}} +_{\text{sup}} y_{\text{sup}})$ avec $+_{\text{inf}}$ l'addition de flottants avec arrondis par défauts, et $+_{\text{sup}}$ celle avec arrondis par excès. Cette stratégie permet un bien meilleur contrôle de la qualité des approximations qui sont faites. Elle peut néanmoins se trouver limitée lorsqu'il s'agit de travailler en dimension plus grande que 1. Elle aboutit aussi souvent à des encadrements trop grossiers.

Depuis quelques décennies, un besoin pour des calculs sur un autre type d'objets ne pouvant être représentés de manière finie et exacte est apparu. Il s'agit des nombres p -adiques (pour p un nombre premier donné), et ce sont ceux-ci qui vont nous intéresser au cours de ce manuscrit. Au premier abord, ils partagent beaucoup de similarités avec les nombres réels. Par exemple, le corps des nombres p -adiques, \mathbb{Q}_p peut se définir de la même manière que celui des réels, \mathbb{R} , comme complétion du corps des rationnels \mathbb{Q} (mais pour une métrique différente de la métrique usuelle sur \mathbb{Q}). De plus, tous les éléments de \mathbb{Q}_p peuvent s'écrire de manière unique sous la forme $\sum_{i \geq l} a_i p^i$, avec $a_i \in \llbracket 0, p-1 \rrbracket$, et en conséquence, pour les mêmes raisons de non-dénombrabilité, presque tout nombre p -adique est non calculable. Cependant, pour ce qui est des calculs, plutôt que l'usage d'une adaptation des flottants, c'est l'équivalent de l'arithmétique d'intervalles qui est le plus souvent utilisé. En effet, ce sont des quantités de la forme $xp^k + O(p^l)$ qui sont d'usage le plus fréquent, avec $x \in \mathbb{N}$ et $O(p^l) = \{y = \sum_{i \geq l} a_i p^i, a_i \in \llbracket 0, p-1 \rrbracket\}$. Dans \mathbb{Q}_p , les $O(p^l)$ sont les boules centrées en zéro, d'où la ressemblance avec l'arithmétique d'intervalle. Par ailleurs, par rapport à cette dernière, les opérations sur les quantités de la forme $xp^k + O(p^l)$ sont beaucoup plus aisées. Par exemple,

$$(xp^k + O(p^l)) + (yp^{k'} + O(p^l)) = (xp^k + yp^{k'}) + O(p^l).$$

Il y a bien sûr un calcul, avec des retenues, à faire pour pouvoir écrire dans ce qui précède $(xp^k + yp^{k'}) + O(p^l)$ sous la forme $zp^{k''} + O(p^l)$. Néanmoins, la remarque importante que nous pouvons faire est que cette somme est connue à la même précision $O(p^l)$, et ce, quelle que soit le calcul de $xp^k + yp^{k'}$. Cette égalité nous permet de dire que les retenus vont du "bon côté" des $O(p^l)$, et ainsi, ne font jamais perdre (en elle-même) de précision. D'autres objets mathématiques intéressants admettent une représentation et un comportement tout à fait semblable, comme par exemple les séries formelles de $\mathbb{Q}[[X]]$ ou de $\mathbb{F}_p[[X]]$. Malgré ces propriétés très agréables dans la gestion de la précision, celle-ci reste (en général) finie⁹, et la plus grande difficulté en précision finie, celle du test à zéro reste présente.¹⁰

Cette thèse s'intéresse à l'étude et aux applications des calculs sur ces objets. En particulier, nous nous intéressons au comportement de la précision lors de calculs non-élémentaires¹¹ ou au cours de l'exécution d'algorithmes. Parmi ceux que nous étudierons figurent le calcul d'une décomposition LU, d'une base de Gröbner, et bien d'autres. Pour cela, nous tâcherons autant que possible d'utiliser

9. Par exemple, on travaille en ne connaissant que les premiers chiffres du développement en base p .

10. Avec plus de détails, il s'agit du fait qu'il n'est pas possible de savoir si un $O(p^l)$ obtenu comme résultat d'un calcul est une approximation de 0 ou d'un nombre qui lui est congru modulo p^l . Bien sûr, augmenter la précision du calcul ne permet pas de régler cette question si la quantité que l'on considère se trouve effectivement être une approximation de zéro.

11. Par opposition à ce que sont, par exemple, addition, multiplication, soustraction et division.

du mieux possible les propriétés très particulières de la topologie, non-archimédienne, qui intervient naturellement lorsque l'on travaille dans un contexte p -adique ou sur des séries formelles.¹²

Un peu d'histoire

Nombres p -adiques

La première définition des nombres p -adiques a maintenant plus d'un siècle, remontant à l'article fondateur de Hensel **Über eine neue Begründung der Theorie der algebraischen Zahlen** [Hen97].¹³ Les motivations de celui-ci étaient déjà de nature algorithmique puisqu'après une définition des nombres p -adiques était déjà introduit le fameux lemme de Hensel¹⁴ pour résoudre des systèmes polynomiaux. Le but était d'obtenir des racines entières à partir de racines modulo p en passant par les nombres p -adiques. Ainsi, les liens très forts des nombres p -adiques avec à la fois l'arithmétique, en étant très proche de \mathbb{Z} et de $\mathbb{Z}/p\mathbb{Z}$, et du calcul numérique transparaissent dès leur première apparition, avec l'objectif clair de combiner ces deux aspects en appliquant des méthodes numériques pour obtenir des résultats sur les entiers.

Depuis, leur étude s'est développée dans de très nombreuses directions. Un des premiers résultats marquant est le théorème de Hasse-Minkowski qui prouve que les formes quadratiques sur les rationnels vérifient un principe local-global, le principe de Hasse : elles admettent un zéro non-trivial sur \mathbb{Q} si et seulement si elles en admettent un dans toutes les complétions de \mathbb{Q} , *i.e.* dans \mathbb{R} et dans tous les \mathbb{Q}_p (p premier).

Dans une autre direction, l'analyse p -adique s'est attachée, avec les travaux de Robba, Amice, Christol et bien d'autres, à partir des années 1970 à l'étude d'objets de nature analytique sur les p -adiques, comme l'étude de fonctions, distributions, équations différentielles, . . . Un des aboutissements de ces méthodes est la preuve de la rationalité des fonctions ζ des variétés définies sur des corps finis par Dwork [Dwo60].

Dans le même temps, les nombres p -adiques sont devenus cruciaux dans le développement de la géométrie arithmétique, où l'on applique des méthodes issues de la géométrie algébrique à des problèmes de nature arithmétique. L'un des accomplissements les plus marquants de ce domaine (et peut-être l'un des plus marquant du vingtième siècle) est la preuve par Deligne des conjectures de Weil, grâce à la construction de bonnes théories cohomologiques à coefficients dans \mathbb{Q}_l , dans [Del74] [Del80], qui étendent le résultat de Dwork précédemment cité. Nous pouvons aussi citer la démonstration du théorème de Fermat-Wiles (anciennement grand théorème de Fermat), dont la preuve utilise de manière cruciale l'étude de certaines représentations galoisiennes p -adiques (voir l'article de Taylor et Wiles [TW95]).

Depuis, les nombres p -adiques ont atteint nombres d'autres domaines des mathématiques, comme la théorie des systèmes dynamiques, la théorie de Lie, ou la cryptographie, et il serait sans doute trop long d'en dresser une liste exhaustive.¹⁵ Parmi ceux-ci, les applications qui vont nous intéresser sont celles qui concernent le calcul et l'algorithmique.

Algorithmique et p -adiques

Dès le commencement de l'avènement de l'algorithmique, le potentiel du calcul sur les p -adiques apparaissait. En effet, ses applications peuvent s'établir naturellement selon deux voies distinctes :

- une première voie est l'usage des nombres p -adiques pour étendre certains calculs effectués sur $\mathbb{Z}/p\mathbb{Z}$ ou \mathbb{Q} à \mathbb{Q}_p , et ainsi, de profiter des bonnes propriétés de \mathbb{Q}_p (caractéristique nulle, complétude, ultramétrie) avant de revenir à $\mathbb{Z}/p\mathbb{Z}$ ou \mathbb{Q} ;
- une seconde voie vient elle de l'existence d'algorithmes qui sont p -adiques par nature. Autrement dit, leur conception est nécessairement sur \mathbb{Q}_p , et n'aurait pas de sens sur $\mathbb{Z}/p\mathbb{Z}$ ou \mathbb{Q} .

12. En particulier, en dimension supérieure à 1, il y a d'autres objets intéressants pour représenter la précision que les boules $O(p^l)$.

13. Certaines idées sur l'étude d'anneaux de valuations discrète étaient cependant déjà présentes un peu plus tôt chez Kummer.

14. Ce dernier est une variante de la méthode de Newton.

15. Et ceci, sans même parler des possibles applications en physique.

Nous présentons ici quelques exemples d’algorithmes utilisant les p -adiques de manière décisive et appartenant à l’une ou l’autre de ces deux catégories. Une remarque importante cependant est que pour chacun de ces algorithmes, qu’il soit ou non purement p -adique, le contrôle et le suivi de la précision y est vital, autant pour ce qui est de la correction (quelle précision en entrée est suffisante pour que le calcul soit correct) que de la complexité (quelle est la plus petite précision à laquelle travailler tout en restant correct) puisque plus la précision est élevée, plus les opérations sont coûteuses.

Passage par les p -adiques

p -adiques et arithmétique élémentaire Dès le début du développement de l’algorithmique effective, accompagnant l’essor du calcul numérique, l’apport possible des nombres p -adiques dans les algorithmes sur les entiers et les corps finis est apparu clair. Déjà dans Knuth [Knu69], l’idée d’effectuer des opérations arithmétiques sur \mathbb{Q}_p à précision finie, puis de revenir dans \mathbb{Q} par un algorithme de reconstruction rationnelle était proposée. L’intérêt étant d’éviter en même temps l’explosion des coefficients que l’on pourrait subir en travaillant dans \mathbb{Q} et les pertes de précision trop importantes qui pourraient apparaître en travaillant dans \mathbb{R} (avec des nombres flottants).

Ces idées ont été raffinées par la suite et appliquées en algèbre linéaire, notamment pour la résolution de systèmes linéaires, par Krishnamurthy, Rao et Subramanian dans [KMRS75] et [Kri75] en 1975. Il y est en particulier discuté du choix $p = 2$ pour l’arithmétique. Par la suite, Dixon a proposé en 1982 dans [Dix82] une méthode de résolution des systèmes linéaires à coefficients entiers qui passe par $\mathbb{Z}/p\mathbb{Z}$ puis \mathbb{Z}_p et, enfin, par un algorithme de reconstruction rationnel. Cette méthode est aujourd’hui encore considérée comme l’une des plus efficaces pour résoudre ce problème.

Quelques années plus tard, Limongelli (et divers co-auteurs) ont proposé d’améliorer ces techniques en parallélisant les calculs : envoyer les rationnels considérés dans plusieurs \mathbb{Q}_p pour différents p , de manière parallèle, puis appliquer le théorème des restes chinois et un algorithme de reconstruction rationnel pour revenir dans \mathbb{Q} . Ceci permet de limiter la taille des calculs à effectuer dans chacun des \mathbb{Q}_{p_i} puisqu’il suffit essentiellement que le produit des précisions $p_i^{n_i}$ sur chaque composante soit assez grand. Un autre avantage de cette technique est de profiter d’une parallélisation très facile puisqu’il n’y a aucune communication nécessaire entre les calculs dans les \mathbb{Q}_{p_i} distincts : seuls les résultats finaux sont à mettre en commun. Citons [Lim93] et [LP94] comme articles présentant cette approche. Une variante modernisée a été proposée dans [LL14] en 2014.

p -adiques et polynômes Au-delà de l’étude des opérations élémentaires et de l’algèbre linéaire, l’intérêt d’utiliser des méthodes comme le lemme de Hensel pour des calculs dans $\mathbb{Q}[X]$ comme la recherche de racine et la factorisation est apparu clair dès 1969 avec l’algorithme de Berlekamp-Zassenhaus (voir [Zas69]). Pour obtenir une factorisation d’un polynôme $f \in \mathbb{Q}[X]$, on débute à partir d’une factorisation modulo p . Ensuite, on la remonte modulo p^n (*i.e.* dans \mathbb{Z}_p à précision n) pour un n assez grand, puis quitte à recombinaison les facteurs obtenus, on en déduit une factorisation sur \mathbb{Q} . La dernière étape n’était pas la plus aisée, et a été simplifiée par Van Hoeij dans [VH02] en 2002.

p -adiques et équations différentielles Une des applications principales des nombres p -adiques en algorithmique est qu’ils permettent, du fait que leur caractéristique est nulle, d’étendre naturellement des opérations qui ne seraient pas possibles sur $\mathbb{Z}/p\mathbb{Z}$. Par exemple, dans $\mathbb{Z}/p\mathbb{Z}[[X]]$, si la dérivation est possible, intégrer X^{p-1} ne l’est pas. Ceci peut être gênant lorsque l’objet sur $\mathbb{Z}/p\mathbb{Z}$ qui nous intéresse est caractérisé par une équation différentielle. La résoudre numériquement sur $\mathbb{Z}/p\mathbb{Z}$ ne paraît pas toujours raisonnable, alors que passer par \mathbb{Z}_p et \mathbb{Q}_p (où l’on peut intégrer sans souci) pour revenir en fin de calcul à $\mathbb{Z}/p\mathbb{Z}$ est souvent plus pratique.

C’est en particulier la voie choisie dans [BGVPS05] où, pour calculer des produits et sommes composées de polynômes, on passe par les sommes de Newton et l’on résout une équation différentielle de la forme $y' = g(X)y$ dans $\mathbb{Z}_p[[X]]$ avant de projeter la solution obtenue numériquement dans $\mathbb{Z}/p\mathbb{Z}[[X]]$.

De manière similaire, pour calculer des isogénies normalisées entre courbes elliptiques, Lercier et Sirvent sont amenés à résoudre dans $\mathbb{Z}_p[[X]]$ une équation différentielle de la forme $y'^2 = g(X)h(y)$ avant de projeter la solution obtenue numériquement dans $\mathbb{Z}/p\mathbb{Z}[[X]]$.

Algorithmes p -adiques par nature

p -adiques et comptage de points Compter le nombre de points qu’une variété a sur un corps fini est l’un des problèmes importants de la géométrie arithmétique. À la suite des techniques développées dans la preuve des conjectures de Weil par Deligne qui lient ce nombre de points à l’action du Frobenius sur de bons espaces de cohomologie et à un polynôme caractéristique associé, de nombreuses méthodes sont apparues pour compter des points. Parmi celles-ci les plus importantes sont celles de Kedlaya [Ked01] et de Lauder [Lau04]. Elles ont pour point commun de devoir à chaque fois effectuer leurs calculs sur des espaces de cohomologie à coefficients p -adiques.

Deux dernières applications Nous n’avons pas nécessairement souhaité être exhaustif dans l’énumération des applications algorithmiques des nombres p -adiques. Cependant, nous pouvons ajouter deux derniers exemples aux algorithmes qui sont p -adiques par nature.

En théorie des codes correcteurs, des codes fondés sur l’usage des p -adiques, particulièrement \mathbb{Z}_2 ont été développés. Nous pouvons citer [CS95] comme article fondateur dans cette direction.

En cryptographie, nous citons une dernière application avec l’usage de relevés en 2-adique pour obtenir des courbes de genre 2 dans [GHW⁺06].

Théorie des bases de Gröbner

Apparues pour la première fois de manière effective dans les travaux de Buchberger (voir [Buc65]), les bases de Gröbner se sont depuis développées jusqu’à pouvoir apporter une réponse à la plupart des questions effectives sur des idéaux dans des anneaux de polynômes. En particulier, calculer une base de Gröbner pour l’ordre lexicographique de l’idéal défini par un système d’équations polynomial permet de ramener la résolution de ce système à celle d’un système (souvent réduit à une seule équation) en une seule variable.

En conséquence, le problème, du calcul effectif d’une base de Gröbner de l’idéal engendré par une famille finie de polynômes se pose naturellement. Dans le cas général, il s’agit d’un problème difficile du point de vue de la complexité, avec des travaux, à la suite de [MM82], montrant notamment une complexité doublement exponentielle en le nombre de variables.

Heureusement, dans des cas génériques, comme celui d’un système donné par une suite régulière, ou de dimension zéro, divers algorithmes comme les algorithmes F5 et F5-Matriciel de Faugère (voir [Fau02] ou [EF14]) ou encore l’algorithme FGLM (de Faugère, Gianni, Lazard et Mora, voir [FGLM93]) permettent d’obtenir une complexité polynomiale en les degrés des entrées. Ces derniers ont permis de rendre le calcul de bases de Gröbner accessibles pour des applications très variées comme la cryptologie, la théorie des codes, la théorie des jeux, les statistiques et bien d’autres.

Cependant, tous les algorithmes que nous venons de citer pour le calcul de bases de Gröbner sont par essence pensés pour le calcul formel (e.g. pour des systèmes définis sur \mathbb{Q} ou sur des corps finis). Leur comportement sur des corps sur lesquels on ne peut travailler qu’à précision finie (réels ou p -adiques) n’est pas évident à prévoir. Ceci n’a pas empêché la conception de plusieurs algorithmes où l’on calcule des bases de Gröbner suivant la première voie d’applications des p -adiques : les calculs sont effectués sur \mathbb{Q}_p avant de revenir à \mathbb{Q} . Nous pouvons citer dans cette voie [Win88], [Arn03] et [RY06]. Remarquons cependant qu’aucun de ces travaux ne s’intéresse au calcul des bases de Gröbner p -adiques pour elles-mêmes.

Géométrie tropicale

De naissance plus récente que les précédents, le dernier domaine qui nous intéresse dans cette thèse est celui de la géométrie tropicale. Il s’intéresse aux problèmes géométriques définis à partir du semi-corps tropical $(\mathbb{R} \cup \{-\infty\}, \min, +)$.¹⁶ Il est possible de définir des polynômes sur $(\mathbb{R} \cup \{-\infty\}, \min, +)$ et ainsi, avec une définition adaptée, des variétés. Celles-ci ont un lien très fort avec les variétés sur des corps valués (dont font partie \mathbb{Q}_p ou \mathbb{Q}_p). En effet, lorsque V est une variété incluse dans $K^{\times n}$ avec K un corps muni d’une valuation val , alors (modulo quelques hypothèses), $val(V)$ est une variété tropicale. On peut imaginer les variétés tropicales comme un pendant plus combinatoire

16. L’adjectif tropical dans le nom vient de la nationalité, brésilienne, d’Imre Simon, fondateur du domaine.

aux variétés classiques. Il se trouve effectivement que, par exemple, les courbes tropicales dans \mathbb{R}^2 sont une union finie de segments et de demi-droites. Le problème de calculer effectivement une représentation de ces variétés tropicales vient alors naturellement. Plusieurs méthodes ont été développées. Dans [BJS⁺07], [Jen] et [Cha13], les auteurs développent des méthodes fondées sur des calculs de bases de Gröbner et d'éventails de Gröbner, tandis que dans [JLY14], il s'agit de méthodes plus numériques, fondées sur la continuation de l'homotopie.

La géométrie tropicale a eu des applications diverses et variées depuis son apparition, notamment en optimisation, en géométrie algébrique énumérative ainsi qu'en géométrie non-archimédienne (voir [MS15] et [Man06]). Récemment une nouvelle motivation pour l'étude de la géométrie tropicale est apparue. En effet, une des voies suivies en théorie du corps à un élément est tropicale, avec notamment la remarque que $(\{-\infty, 0\}, \min, +)$ est le seul semi-corps fini qui ne soit pas un corps (voir par exemple [CC15], [Con15]).

Contributions

Deux axes structurent les travaux que nous présentons dans cette thèse. Le premier a trait au suivi de la précision dans un contexte p -adique, d'une manière théorique comme pratique. Le second lui concerne le calcul de bases de Gröbner dans un contexte p -adique, et étudie la question suivante : quels calculs de bases de Gröbner sont possibles sur \mathbb{Q}_p ?

Précision p -adique

Présentation de la problématique

La gestion de la précision est un problème central dans toutes les méthodes et algorithmes utilisant des p -adiques, que ces derniers soient utilisés pour étendre des calculs sur \mathbb{Q} ou $\mathbb{Z}/p\mathbb{Z}$ ou soient p -adiques par essence. Ceci est inhérent à la nature nécessairement non finiment représentable des éléments de \mathbb{Q}_p . Jusqu'à présent, lorsqu'on souhaitait suivre la précision lors d'un calcul ou au cours d'un algorithme, la méthode principale était d'utiliser des formules du type de

$$(a + O(p^n)) + (b + O(p^m)) = (a + b) + O(p^{\min(n,m)}).$$

Si celles-ci sont optimales lorsqu'on les applique pour une seule opération élémentaire, en appliquer plusieurs consécutivement conduit en général à des estimation de pertes de précision plus grandes que de raison. Elles peuvent parfois être compensées lorsque l'on dispose d'*invariants* de l'algorithme, comme par exemple le fait que tous les nombres considérés sont des entiers ou que les vecteurs en sortie sont dans un certain sous-module dont on aurait une présentation à précision infinie. Hélas, tous les algorithmes ne possèdent pas nécessairement de tels invariants. De plus, suivre un algorithme en appliquant de telles formules produit une estimation sur la précision du résultat en sortie qui dépend directement de l'algorithme suivi. Ainsi, lorsqu'un objet peut être calculé de différentes manière, cette méthode ne donne pas de réponse définitive sur ce qui serait optimal du point de vue de la perte de précision.

Une première réponse théorique

Avec Xavier Caruso et David Roe, nous fournissons un résultat donnant de manière claire le comportement qualitatif de la précision dans un contexte où le calcul que l'on fait est celui d'un objet qui dépend de manière suffisamment régulière des entrées.

Pour cela, nous définissons la notion de réseau de précision. Il s'agit dans un contexte p -adique¹⁷, en particulier en dimension plus grande que 1, de remplacer un suivi coordonnée par coordonnée par un suivi global avec des réseaux.

Par exemple, dans \mathbb{Q}_p^d , nous considérons tous les \mathbb{Z}_p -réseaux de \mathbb{Q}_p^d et non seulement les données de précision de la forme $(O(p^{n_1}), \dots, O(p^{n_d}))$, qui constituent les réseaux diagonaux.¹⁸ L'avantage décisif de ce choix est que l'ensemble des \mathbb{Z}_p -réseaux de \mathbb{Q}_p^d est stable par action de $M_n(\mathbb{Q}_p)$. En réel,

17. Le contexte d'un espace de Banach ultramétrique fonctionnerait aussi.

18. Remarquons en particulier que les boules centrées en zéro sont bien sûr des réseaux.

plutôt que des réseaux, cela correspondrait à regarder des ellipsoïdes au lieu de prendre seulement des boules. Cependant, le résultat suivant montre que le comportement des réseaux de \mathbb{Q}_p^d sous l'action d'applications différentiables les rend bien plus pratiques que les ellipsoïdes. En voici un énoncé :

Théorème A (Lemme 2.2.4). *Soit $n, m \in \mathbb{N}^*$ et $f : U \rightarrow \mathbb{Q}_p^m$ une application définie sur un ouvert U de \mathbb{Q}_p^n . Supposons que f est différentiable¹⁹ en un point $v_0 \in U$ et que la différentielle $f'(v_0)$ est surjective.*

Alors, pour tout $\rho \in]0, 1]$, il existe un réel positif δ tel que, pour tout $r \in]0, \delta[$, pour tout réseau H compris entre les boules $B_{\mathbb{Q}_p^n}(0, \rho r)$ et $B_{\mathbb{Q}_p^n}(0, r)$, nous avons :

$$f(v_0 + H) = f(v_0) + f'(v_0) \cdot H.$$

Ainsi, le comportement de la précision au voisinage de v_0 est entièrement dicté par $f'(v_0)$. En particulier, si H est un réseau vérifiant les conditions de l'hypothèse, alors $f'(v_0) \cdot H$ est aussi un réseau (qui reste ouvert).

Ceci permet de donner une réponse claire au comportement de la précision pour le calcul d'objets définis par une fonction différentiable (dont sont les fonctions polynomiales).

Par exemple, en calculant la dérivée de la fonction qui à M associe sa décomposition LU , nous pouvons donner explicitement le comportement de la précision sur la décomposition LU d'une matrice M étant donnée une précision initiale sur M . Ceci donne un résultat qui est indépendant de l'algorithme que l'on utiliserait pour calculer cette décomposition LU .

Le résultat précédent ne donne que le comportement de la précision de manière asymptotique et qualitative : il ne donne pas de borne explicite à partir de laquelle un réseau H est assez petit pour vérifier $f(v_0 + H) = f(v_0) + f'(v_0) \cdot H$. Cependant, ce problème peut être résolu de manière effective lorsque la fonction f qui nous intéresse est analytique en v_0 . En effet, si f s'écrit $f(v_0 + h) = \sum_{n \geq 0} f_n(h)$, avec f_n polynôme homogène de degré n , au voisinage de v_0 , nous avons alors le résultat suivant :

Théorème B (Corollaire 2.3.12). *Soit $f : \mathbb{Q}_p^n \rightarrow \mathbb{Q}_p^m$ s'écrivant $f(v_0 + h) = \sum_{n \geq 0} f_n(h)$ au voisinage de v_0 comme précédemment. Nous supposons donné un $C \in \mathbb{R}_{>0}$ tel que $B_{\mathbb{Q}_p^m}(0, 1) \subset f_1(B_{\mathbb{Q}_p^n}(C))$. Notons $NP((f_n)_{n \geq 2})$ le polygone de Newton défini par les points $(n, -\log \|f_n\|)$, pour $n \geq 2$ et soit $NP((f_n)_{n \geq 2})^*$ sa transformée de Legendre. Soit ν tel que :*

$$NP((f_n)_{n \geq 2})^*(\nu) < \nu - \log(C).$$

Alors la conclusion du Théorème A est satisfaite pour $\delta = e^\nu$.

Ainsi, il est possible de savoir précisément quand est-ce que nous sommes dans la situation où $f(v_0 + H) = f(v_0) + f'(v_0) \cdot H$: il suffit pour cela de calculer les normes des f_n dans le développement en série de f en v_0 .²⁰ Nous appliquons ces résultats à divers exemple comme le calcul de la décomposition LU ou le calcul des termes de la suite récurrente SOMOS-4, suite récurrente apparaissant dans le calcul de la suite $(P + nQ)_{n \in \mathbb{N}}$ où P et Q sont des points sur une certaine courbe elliptique.

Par ailleurs, les résultats précédents se généralisent très naturellement au cas de variétés sur \mathbb{Q}_p . Ceci peut être agréable lorsqu'on travaille sur des variétés projectives ou algébriques telles que des courbes elliptiques.

Nous appliquons les méthodes précédentes sur divers exemples en algèbre linéaire ainsi que sur les polynômes univariés. Nous fournissons quelques applications numériques montrant que l'utilisation des réseaux permet un bien meilleur contrôle de la précision que les méthodes directes. Par exemple, pour un produit de mille matrices carrées 2×2 aléatoires à coefficient dans \mathbb{Z}_p , la perte de précision constatée par des méthodes directes est en moyenne d'environ 160 chiffres significatifs, contre seulement 8 en utilisant un suivi de précision par les réseaux.

19. Voir la Définition 2.2.1 pour ce que nous appelons précisément être différentiable.

20. En fait, une majoration peut suffire.

Une seconde réponse pratique

Les résultats précédents permettent, lorsqu'on s'intéresse à un calcul ou au résultat d'un algorithme, de savoir numériquement et précisément quelle est la précision définie sur la sortie par la précision sur les entrées. Cependant, ils ne donnent pas de manière d'atteindre dans les faits cette précision théorique sur la sortie. Souvent, en effet, des gains de précision ne sont pas constatés lorsque le calcul est implémenté en machine et effectué à précision finie. Dans ce cas, la précision obtenue directement en pratique est nettement moins bonne que celle que les résultats précédents donnent.

Ceci est particulièrement visible sur l'exemple du calcul des termes de la suite récurrente SOMOS-4. Il est facile, vu les résultats connus sur cette suite, de montrer que la connaissance de ses termes initiaux à précision $O(p^n)$ définit tous ses termes à la même précision. Pourtant, une implémentation naïve en machine ne le fait pas apparaître, perdant rapidement tous les chiffres de précision.

Nous proposons une méthode, dite adaptative, pour atteindre le comportement de la précision prédit par le Théorème A. Celle-ci repose sur l'idée suivante : si le calcul en machine perd des chiffres de précision de manière indue (à cause de gains ou de simplifications que ce calcul ne peut faire apparaître), il suffit d'ajouter avant chaque opération suffisamment de chiffres de précision, arbitraires, pour absorber la perte de précision indue donnée par cette opération. Ces chiffres de précision arbitraire servent, en quelque sorte, de tampon pour absorber la perte de précision indue.

Grâce au fait que nous nous plaçons dans un contexte ultramétrique, ces chiffres de précisions arbitraires (sans lien avec les données initiales dont sont données des approximations, et qui seront lorsque possible pris tous nuls) ne modifient pas le résultat final une fois réduit à la précision donnée par le Théorème A.

Pour donner une première idée, nous donnons l'exemple suivant : lors du calcul du déterminant d'une matrice de $M_n(\mathbb{Z}_p)$ dont tous les coefficients sont donnés à la même précision $O(p^l)$, on sait par l'expression explicite du déterminant en fonction des coefficients que celui-là est dans \mathbb{Z}_p et est connu au moins à cette même précision $O(p^l)$. Cependant, appliquer un calcul par la méthode de Gauss entraîne une perte de précision due aux divisions par les pivots. Une manière d'appliquer une méthode adaptative²¹ serait d'augmenter arbitrairement la précision de $O(p^l)$ à $O(p^{l+m})$ où m est une majoration de la valuation du déterminant de la matrice considérée (ce qui peut se lire directement sur les coefficients dans ce contexte ultramétrique).

L'utilisation combinée de l'étude théorique différentielle et d'une telle méthode adaptative permet d'obtenir la précision théorique optimale donnée par le Théorème A. C'est par exemple le cas pour le calcul des termes de la suite SOMOS-4.

Un exemple complet

Enfin, nous appliquons et illustrons les résultats et méthodes précédents sur plusieurs exemples complets. Nous avons déjà évoqué la suite récurrente SOMOS-4.

Notre autre exemple majeur est celui de la résolution d'équations différentielles de la forme $y' = f(x)h(y)$. Il s'agit d'un travail en commun avec Pierre Lairez. Soit $f, h \in \mathbb{Z}_p[[t]]$. Nous nous intéressons à l'équation différentielle suivante :

$$\begin{cases} y' = g \cdot h(y), \\ y(0) = 0. \end{cases} \quad (\text{A})$$

Supposons que g et h sont telles que l'équation précédente ait une solution dans $\mathbb{Z}_p[[t]]$. En appliquant les Théorèmes A, B et une méthode adaptative, nous obtenons le résultat suivant :

Théorème C (Théorème 5.1.2). *Soit m et n deux entiers positifs, tels que $m \geq 2 \log_p(n+1) + \frac{2}{p-1}$. Notons E la fonction partie entière sur \mathbb{Z} . Alors, étant donnés des approximations modulo (p^m, t^n) de g et h , on peut calculer une approximation modulo $(p^{m-E(\log_p(n))}, t^{n+1})$ de la solution y de (A).*

De plus nous donnons une estimation précise de la complexité nécessaire à ce calcul.

Ce résultat donne une nouvelle interprétation de l'estimation de perte de précision dans le cas particulier où (A) est une équation différentielle linéaire, qui est traité dans [BGVPS05] et [GvdHL15]. Il améliore aussi les résultats connus sur le cas de l'équation différentielle $y'^2 = g \cdot h(y)$, pour $p \neq 2$, qui est traité dans [LS08].

21. Celle-ci est légèrement naïve, et nous en verrons plus sur le déterminant en Sous-Section 3.2.2.

Systèmes polynomiaux

Présentation de la problématique

À la question de savoir quelles bases de Gröbner peuvent être calculées à précision finie dans $A = \mathbb{Q}_p[X_1, \dots, X_n]$, il est possible de donner une première réponse très directe : pour $F = (f_1, \dots, f_s)$ des polynômes de A connus à une précision $O(p^N)$, très supérieure aux valuations des coefficients non nuls qui apparaissent lors du calcul d'une base de Gröbner de $\langle F \rangle$ par un algorithme donné, et que l'on néglige tous les termes de la forme $O(p^k)X^\alpha$ qui apparaîtraient dans le calcul, alors l'exécution de l'algorithme choisi serait la même qu'à précision infinie, et l'on obtiendrait bien une approximation d'une base de Gröbner de $\langle F \rangle$.

Néanmoins, ceci n'est pas satisfaisant pour plusieurs raisons. Tout d'abord, ce raisonnement ne donne aucun contrôle sur la précision nécessaire pour que le calcul se passe effectivement comme à précision infinie. Celle-ci dépend des polynômes rencontrés au cours de l'algorithme, et il n'est pas évident d'estimer leurs coefficients *a priori*. Plus grave, en suivant une telle méthode où l'on néglige les $O(p^k)X^\alpha$, même si la précision en entrée est très grande devant celle qui serait nécessaire pour assurer que le calcul se passe bien comme en précision infinie, il n'est pas directement possible d'assurer que le calcul s'est effectivement effectué de cette manière. En effet, lorsque la précision est finie, il n'est pas possible de déterminer si, en supposant que $X^\alpha > X^\beta$ pour l'ordre monomial considéré,

$$((1 + O(p^n))X^\alpha + (2 + O(p^n))X^\beta) - ((1 + O(p^n))X^\alpha) = O(p^n)X^\alpha + (2 + O(p^n))X^\beta$$

a pour monôme de tête X^α ou X^β . À précision finie, il n'est pas possible de décider si la précision est insuffisante pour montrer si $O(p^n)X^\alpha$ est bien nul, dû à une compensation entre coefficients, auquel cas le terme de tête est $(2 + O(p^n))X^\beta$ ou si la précision manque pour montrer que son coefficient est non nul, et le terme de tête devrait être $O(p^n)X^\alpha$. Il s'agit du bien connu problème en calcul numérique du *test à zéro*.

Hélas, de telles compensations²² sont monnaie courante lors de calcul de bases de Gröbner. Il suffit de considérer l'algorithme de division par une famille de polynôme (g_1, \dots, g_t) : si le reste est nul à précision infinie, lors d'un calcul à précision finie, on n'obtiendra en général que des $O(p^k)X^\alpha$ avec X^α non divisible par un $LM(g_i)$.

En conclusion, est-on alors condamné à ne pouvoir qu'espérer que la précision sur nos calculs est suffisante pour distinguer les zéros qui apparaîtraient à précision infinie des coefficients non nuls ? Est-on ainsi condamné à ne pas pouvoir certifier quels sont les monômes de tête ? Nous allons voir que dans des cas particuliers, qui peuvent être génériques ou conjecturalement génériques, il est possible de fournir une réponse satisfaisante.

Approche directe

Afin de répondre à la problématique précédente, nous nous intéressons à l'application $(f_1, \dots, f_s) \mapsto (g_1, \dots, g_t)$ qui à une famille de s polynômes de $\mathbb{Q}_p[X_1, \dots, X_n]$ associe la base de Gröbner réduite de $\langle f_1, \dots, f_s \rangle$, pour un ordre monomial donné ω , et étudions leur lieu de continuité et de différentiabilité.

Notons A_d l'espace vectoriel des polynômes homogènes de degré d dans $\mathbb{Q}_p[X_1, \dots, X_n]$. Nous définissons un ouvert de Zariski $U \subset A_{d_1} \times \dots \times A_{d_s}$ formé des s -uplets qui sont une suite régulière et qui vérifient une hypothèse supplémentaire de généricité par rapport à ω ²³. Cette ouvert est dense lorsque ω est l'ordre grevlex si la conjecture de Moreno-Socias est vérifiée. En étudiant une adaptation de l'algorithme F5-Matriciel de Faugère (voir [Fau02], [BFS14]), nous montrons le résultat suivant :

Théorème D (Proposition 7.3.1). *Soit $a \in U$. Il existe un voisinage ouvert V_a de a ,²⁴ et d et r tels que l'application qui à $v \in V_a$ associe la base de Gröbner réduite de $\langle v \rangle$ pour ω s'écrive $\Psi : V_a \rightarrow A_{\leq d}^r$ et soit différentiable. De plus, son module de continuité est contrôlé par les mineurs des matrices de Macaulay définies par a .*

22. En anglais, on utiliserait certainement *cancellation*.

23. Pour être précis, il s'agit que les idéaux $\langle f_1, \dots, f_i \rangle$ sont faiblement- ω .

24. V_a est défini explicitement par les mineurs des matrices de Macaulay définies par a jusqu'au degré $1 + \sum_{i=1}^s d_i - 1$.

Il est ainsi possible de calculer une base de Gröbner approchée de $a + O(p^N)$ dès que N est assez grand pour que $B(0, \frac{1}{p^N}) \subset V_a$ par une adaptation de l'algorithme F5-Matriciel. Sur un tel V_a , il nous est aussi possible de calculer explicitement Ψ' :

Théorème E (Théorème 7.3.2). *Soit $M \in \mathbb{Q}_p[X_1, \dots, X_n]^{s \times r}$ tel que $\Psi(a) = a \times M$. Alors, pour tout $\delta f \in A_{d_1} \times \dots \times A_{d_s}$, nous avons :*

$$\Psi'(a) \cdot \delta f = \delta f \times M \mod \Psi(f).^{25}$$

Enfin, les résultats précédents s'étendent naturellement au cas affine : si l'ordre \leq considéré raffine le degré, et si (f_1, \dots, f_s) sont tels que leurs composantes homogènes de plus haut degré (f_1^h, \dots, f_s^h) sont dans l'ouvert $U \subset A_{d_1} \times \dots \times A_{d_s}$, alors $LM(\langle f_1^h, \dots, f_s^h \rangle) = LM(\langle f_1, \dots, f_s \rangle)$ et ainsi, il est possible de calculer une base de Gröbner approchée de (f_1, \dots, f_s) en appliquant ce qui précède à (f_1^h, \dots, f_s^h) .

Changement d'ordre

La définition de l'ouvert précédent requiert des conditions difficiles à contrôler. Elles ne sont au mieux que conjecturalement générique pour l'ordre grevlex, et parfois vides pour certains ordres monomiaux (dont l'ordre lexicographique). Dans ce cas, l'étude précédente montre qu'un calcul direct par l'algorithme F5-Matriciel n'est pas stable au sens où il n'y est pas possible de discerner les termes de têtes (pour un tel ordre) à précision finie. Cependant, si (f_1, \dots, f_s) est tel que'il est possible d'en calculer une base de Gröbner par un algorithme F5-Matriciel pour l'ordre grevlex, alors est-ce qu'un algorithme de changement d'ordre monomial pourrait permettre d'en déduire une base de Gröbner pour l'ordre lexicographique ?

Avec Guénaél Renault, nous fournissons une réponse positive à cette question lorsque l'idéal concerné est de dimension zéro grâce à une étude puis une adaptation de l'algorithme FGLM. Nous obtenons le résultat suivant :

Théorème F (Théorème 8.1.6). *Soit G une base de Gröbner approchée à précision $O(p^N)$ d'un idéal I de $A = \mathbb{Q}_p[X_1, \dots, X_n]$ pour un ordre monomial \leq . Supposons que l'idéal I est de dimension 0 et de degré δ . Soit \leq_2 un second ordre monomial. Soit M la matrice de changement de base entre les bases canoniques de A/I pour \leq et \leq_2 , et soit σ_δ la plus grande valuation d'un de ses facteurs invariants.*

Alors, si $N > \sigma_\delta$, nous pouvons calculer, par un algorithme FGLM, une base de Gröbner approchée de I pour \leq_2 , et ses coefficients seront connus à précision $O(p^{N-2\sigma_\delta})$ au moins. Le temps de calcul est en $O(n\delta^3)$ opérations sur \mathbb{Q}_p à précision N .

Lorsque l'on souhaite passer de grevlex à lex et que l'idéal I admet une base de Gröbner avec variables en position générale²⁶, il est possible d'adapter les variantes plus rapides de l'algorithme FGLM (issues, entre autres, de [FM11] et [FGR14]) pour obtenir la même perte de précision, mais un temps de calcul en $O(n\delta^2) + O(\delta^3)$ opérations dans \mathbb{Q}_p , à précision N .

Une approche tropicale

Lorsqu'on s'intéresse aux problèmes de géométrie tropicale sur des corps non trivialement valués, une notion d'ordre sur les termes et de base de Gröbner tropicale prenant en compte la valuation du corps apparaît naturellement. Par exemple, dans $A = \mathbb{Q}_p[X_1, \dots, X_n]$, si $w \in \mathbb{R}^n$, il est possible d'ordonner les termes de A en comparant, pour $c_\alpha X^\alpha$, les $val(c_\alpha) + \sum_{i=1}^n w_i \alpha_i$. L'hypersurface tropicale définie par un $f \in A$ est alors l'adhérence des points $w \in \mathbb{R}^n$ tel que le maximum parmi les termes de f , selon l'ordre sur les termes défini par w , est atteint par au moins deux des termes de f . Ceci s'étend à d'autres variétés que des hypersurfaces via une définition naturelle de base de Gröbner tropicale qui s'obtient après ajout d'un ordre monomial classique pour briser les égalités entre termes pour l'ordre précédent.

Chan et Maclagan ont décrit un algorithme, adapté de l'algorithme de Buchberger, dans [CM13] pour calculer de telles bases de Gröbner tropicales pour des idéaux homogènes.

25. Ici, le \mod est le reste dans la division par $\Psi(a)$

26. *shape position* est le terme en anglais

Nous montrons que, dans le même contexte, après quelques adaptations sur la manière de réduire les matrices, le critère F5 et un algorithme F5-Matriciel tropical sont applicables pour calculer des bases de Gröbner tropicales.

De plus, lorsque l'on doit travailler à précision finie, nous avons le résultat suivant :

Théorème G. *Soit $(d_1, \dots, d_s) \in \mathbb{N}^{*s}$ et $A = \mathbb{Q}_p[X_1, \dots, X_s]$. Soit $w \in \mathbb{R}^n$ et \leq un ordre monomial. Soit U l'ouvert de $A_{d_1} \times \dots \times A_{d_s}$ formé des suites régulières. Alors, si $a \in U$, il existe un voisinage V_a de a dans U ²⁷, tel qu'un algorithme F5-Matriciel tropical calcule des bases de Gröbner tropicales approchées de a .*

Remarquons que l'ouvert U est dense dans $A_{d_1} \times \dots \times A_{d_s}$ et toujours strictement plus gros que celui qui apparait dans le Théorème D. Ainsi, calculer des bases de Gröbner tropicales approchées est plus facile lorsque la valuation est non triviale que calculer des bases de Gröbner, au sens que seul la régularité des polynômes en entrée et une précision suffisante suffisent.

En outre, si nos motivations ne sont pas forcément de nature tropicale, mais plutôt de choisir le poids w pour calculer une base de Gröbner tropicale avec le moins de perte de précision, nous recommandons le poids $w = (0, \dots, 0)$ qui au moins sur les exemples que nous avons considéré, est celui qui entraîne le moins de perte de précision pour l'algorithme F5-Matriciel tropical.²⁸

Enfin, lorsque l'on s'intéresse à des idéaux de dimension zéro, il est naturel de se demander s'il est possible de calculer une base de Gröbner à partir d'une base de Gröbner tropicale. Avec Guénaël Renault, nous avons alors le résultat suivant :

Théorème H (Propositions 9.3.13 et 9.3.14). *Soit G une base de Gröbner tropicale approchée à précision $O(p^N)$ d'un idéal homogène I de $A = \mathbb{Q}_p[X_1, \dots, X_n]$ pour un ordre sur les termes \leq donné par un poids w et un ordre monomial \leq_1 . Supposons que l'idéal I est de dimension 0 et de degré δ . Soit \leq_2 un ordre monomial classique. Soit M la matrice de changement de base entre les bases canoniques de A/I pour \leq et \leq_2 , et soit σ_δ la plus grande valuation d'un de ses facteurs invariants.*

Alors, si $N > \sigma_\delta$, nous pouvons calculer, par un algorithme FGLM, une base de Gröbner approchée de I pour \leq_2 , et ses coefficients seront connus à précision $O(p^{N-2\sigma_\delta})$ au moins. Le temps de calcul est en $O(n\delta^3)$ opérations sur \mathbb{Q}_p à précision N .

Ceci nous amène à nos deux derniers résultats pour conclure sur le calcul de bases de Gröbner en dimension zéro :

Théorème I (Proposition 9.3.13). *Soit $F = (f_1, \dots, f_n)$ une suite régulière de polynômes homogènes de $A = \mathbb{Q}_p[X_1, \dots, X_n]$. Soit \leq un ordre monomial sur A . Alors si la précision sur les coefficients des f_i est assez grande²⁹, il est possible de calculer une base de Gröbner de F pour \leq en suivant la méthode suivante :*

1. Calculer une base de Gröbner G_{trop} de $\langle F \rangle$ pour l'ordre tropical donné par $w = (0, \dots, 0)$ et \leq grâce à un algorithme F5-Matriciel tropical;
2. Calculer une base de Gröbner G de $\langle F \rangle$ pour \leq grâce à G_{trop} et un algorithme FGLM.

Lorsqu'on ne se trouve pas dans le cas homogène, nous avons finalement le résultat suivant :

Théorème J (Théorème 9.4.1). *Soit $F = (f_1, \dots, f_n)$ des polynômes de $A = \mathbb{Q}_p[X_1, \dots, X_n]$ tels que leurs composantes homogènes de plus haut degré $F^h = (f_1^h, \dots, f_n^h)$ forme une suite régulière de polynômes homogènes de $A = \mathbb{Q}_p[X_1, \dots, X_n]$. Soit \leq un ordre monomial sur A . Supposons que \leq ne raffine pas le degré. Alors si la précision sur les coefficients des f_i est assez grande³⁰, il est possible de calculer une base de Gröbner de F en suivant la méthode suivante :*

1. Calculer une base de Gröbner G_{trop}^h de $\langle F^h \rangle$ pour l'ordre tropical donné par $w = (0, \dots, 0)$ et \leq grâce à un algorithme F5-Matriciel tropical;

27. V_a est donné par les mineurs des matrices de Macaulay définies par a .

28. Il est aussi celui pour lequel nous pouvons donner les meilleures bornes sur les pertes de précision.

29. Ce qui est donné par des mineurs des matrices de Macaulay définies par F et les facteurs invariants d'une matrice de changement de base de $A/\langle F \rangle$.

30. Ce qui est donné par des mineurs des matrices de Macaulay définies par F^h , les facteurs invariants de matrice de changement de base de $A/\langle F^h \rangle$ et $A/\langle F \rangle$, et les valuations des termes de tête de G_0 .

2. Calculer une base de Gröbner G_0^h pour grevlex de $\langle F^h \rangle$ grâce à G_{trop}^h et un algorithme FGLM;
3. Grâce à l'écriture des éléments de G_0^h en fonction de ceux de F^h ³¹, en déduire une base de Gröbner G_0 de $\langle F \rangle$ pour grevlex;
4. Grâce à l'algorithme FGLM, en déduire une base de Gröbner de $\langle F \rangle$ pour \leq .

Lorsque \leq raffine le degré, il est possible de remplacer grevlex par \leq et de s'arrêter à l'étape 3.

Ceci montre en particulier la continuité (et même la différentiabilité) du calcul de bases de Gröbner en dimension zéro au voisinage de polynôme dont les composantes homogènes de plus haut degré forment une suite régulière, ce qui est une condition Zariski-ouverte et non vide.

Publications

L'essentiel des Chapitres 2 et 4, qui est un travail en commun avec Xavier Caruso et David Roe, a fait l'objet d'une publication lors de la Conférence ANTS XI, dans le *LMS Journal of Computation and Mathematics, Volume 17 - Special Issue A (Algorithmic Number Theory Symposium XI) - 2014*, [CRV14].

Les Sections 3.2 et 3.3 du Chapitre 3, travail en commun avec Xavier Caruso et David Roe, ont fait l'objet d'une publication en un article lors de la conférence ISSAC 2015, [CRV15].

La majeure partie du contenu du Chapitre 7 a fait l'objet d'une publication lors de la conférence ISSAC 2014, [Vac14].

Les Sections 9.1 et 9.2 du Chapitre 9 ont fait l'objet d'une publication en un article lors de la conférence ISSAC 2015, [Vac15].

31. Ceci peut s'obtenir grâce aux polynômes G_{trop}^h et leur écriture en fonction de F^h .

Organisation du manuscrit

Ce manuscrit est divisé en deux parties, selon les deux axes de cette thèse : l'étude de la précision différentielle, puis celle des systèmes polynomiaux, et des bases de Gröbner correspondantes, sur un corps complet discrètement valué à précision finie.

Première partie : Précision différentielle

Chapitre 1 Ce chapitre s'attache à exposer les méthodes classiques de suivi de la précision sur des corps complets discrètement valués à précision finie, tel que \mathbb{Q}_p ou $\mathbb{Q}[[T]]$. Nous montrons que la méthode, dite directe, où l'on attache des $O(\pi^k)$ à chaque coefficient, permet de traiter plusieurs calculs en algèbre linéaire, comme l'échelonnement en lignes ou le calcul de la forme normale de Smith. Néanmoins, nous illustrons que cette méthode n'est vraiment pas toujours satisfaisante.

Chapitre 2 Dans ce chapitre, nous introduisons l'outil essentiel de la précision différentielle, les réseaux, et avec cet outil, nous montrons les Théorèmes A et B. Enfin, nous généralisons ce résultat au cadre des variétés.

Chapitre 3 Ce chapitre s'intéresse à diverses applications des Théorèmes A et B, principalement en algèbre linéaire et sur les polynômes. Nous traitons explicitement ces exemples et donnons, autant que possible, le comportement qualitatif et quantitatif de la précision grâce aux méthodes du chapitre précédent.

Chapitre 4 Dans ce chapitre, nous présentons une méthode pour atteindre en pratique les comportements sur la précision prédits par les méthodes précédentes. Il s'agit de compenser les pertes de précisions indues causées par un calcul en machine effectué de manière trop directe.

Chapitre 5 Ce dernier chapitre de la première partie traite d'un exemple, celui de la résolution de certaines équations différentielles sur $\mathbb{Q}_p[[t]]$, de manière complète : calcul de la dérivée, estimation de la précision nécessaire pour que son comportement soit dicté par le premier ordre, mise en place d'une méthode adaptative pour atteindre ce comportement.

Seconde partie : Systèmes polynomiaux

Chapitre 6 Nous débutons cette seconde partie par un chapitre fournissant une brève introduction à la théorie des bases de Gröbner, ainsi que la présentation des différents algorithmes pour les calculer qui nous intéresseront dans les chapitres suivants : F5-Matriciel, FGLM, et leurs diverses variantes.

Chapitre 7 Ce chapitre s'intéresse au calcul direct d'une base de Gröbner lorsque la précision est finie. Il donne une réponse positive, ainsi qu'un algorithme adapté de l'algorithme F5-Matriciel, sur une partie ouverte des systèmes de polynômes ayant degrés bornés et n'étant pas sur-déterminés. Celle-ci est conjecturalement dense pour grevlex.

Chapitre 8 Dans ce chapitre, nous étudions l'algorithme FGLM de changement d'ordre monomial lorsque la précision est finie, et montrons que, à quelques adaptations près, celui-ci est stable.

Chapitre 9 Afin de palier à la non-densité des ouverts vus en Chapitre 7 sur lesquels les bases de Gröbner peuvent être calculées à précision finie, et aussi pour des raisons de géométrie tropicale, nous nous intéressons aux bases de Gröbner tropicales. Nous montrons que celles-ci peuvent bien être calculées par une adaptation de l'algorithme F5-Matriciel, sur des ouverts denses. Nous montrons ensuite que lorsque l'on est en dimension zéro, il est possible d'appliquer un algorithme FGLM pour passer d'une base de Gröbner tropicale à une base de Gröbner classique. Ceci nous permet de clore notre étude.

Première partie

Précision différentielle

Résumé

Dans cette partie, nous présentons un nouveau modèle pour étudier et suivre la précision dans un contexte p -adique, qui consiste, plutôt qu'à regarder des $O(p^n)$ sur chaque coordonnée, à considérer des \mathbb{Z}_p -réseaux dans \mathbb{Q}_p^d . Nous avons alors le résultat suivant : étant donnée une approximation $x + H$ d'un élément $x \in \mathbb{Q}_p^d$ par un réseau H , assez petit, la précision sur $f(x + H)$ est essentiellement donnée par l'approximation de Taylor de f au premier ordre, $f(x + H) = f(x) + f'(x) \cdot H$. Ceci permet de ramener le suivi de la précision de manière qualitative (ou asymptotique) à un simple calcul de différentielle. De plus, en connaissant les normes des dérivées d'ordre supérieur de f , nous pouvons préciser de manière effective quand le réseau H est assez petit.

Des calculs naïfs en machine sur les p -adiques ne permettent pas toujours d'observer en pratique la précision prédite par une estimation au premier ordre. Pour y remédier, nous proposons une méthode, dite méthode adaptative, pour atteindre ce comportement de la précision donné par l'approximation au premier ordre.

Nous illustrons ces deux méthodes sur divers exemples : opérations sur les polynômes, opérations sur les matrices et en algèbre linéaire. Enfin, nous proposons un exemple d'application traité de manière complète : l'étude de la résolution numérique de certaines équations différentielles p -adiques à variables séparables.

Notations

Cadre général

Soit K un corps muni d'une valeur absolue $|\cdot| : K \rightarrow \mathbb{R}_{\geq 0}$. Nous supposons que la métrique induite est ultramétrique (*i.e.* $|x + y| \leq \max(|x|, |y|)$) et que K est complet pour cette métrique. Par exemple, nous pouvons prendre $K = \mathbb{Q}_p$ avec la valeur absolue p -adique ou $K = k((t))$ avec valeur absolue t -adique. Nous notons \mathcal{O}_K pour l'anneau $\{x \in K : |x| \leq 1\}$ et supposons que K contient un sous-anneau dense effectif $R_{\text{eff}} \subset K$ [Rab60]. Cette hypothèse est vérifiée pour $K = \mathbb{Q}_p$ et $K = \mathbb{F}_p((t))$ en prenant $R_{\text{eff}} = \mathbb{Z}[\frac{1}{p}]$ ou $R_{\text{eff}} = \mathbb{Q}$ dans le cas où \mathbb{Q}_p et $R_{\text{eff}} = \mathbb{F}_p[t, t^{-1}]$ ou $R_{\text{eff}} = \mathbb{F}_p(t)$ pour $\mathbb{F}_p((t))$.

Si E est un K -espace vectoriel, possiblement de dimension infinie, alors une *norme ultramétrique* sur E est une application $\|\cdot\| : E \rightarrow \mathbb{R}^+$ satisfaisant :

- (i) $\|x\| = 0$ si et seulement si $x = 0$;
- (ii) $\|\lambda x\| = |\lambda| \cdot \|x\|$;
- (iii) $\|x + y\| \leq \max(\|x\|, \|y\|)$.

Un K -espace de Banach est K -espace vectoriel normé complet. Nous pouvons remarquer que tout K -espace vectoriel normé de dimension finie est automatiquement complet et toutes les normes sur cette espace sont équivalentes. Un *réseau* dans un K -espace de Banach E est un sous- \mathcal{O}_K -module de E ouvert borné. Nous soulignons que tout réseau H de E est aussi fermé puisque son complémentaire est l'union des ensembles $a + H$ (avec $a \notin H$) qui sont tous ouverts. Pour un K -espace de Banach E et $r \in \mathbb{R}_{\geq 0}$, nous noterons les boules de rayon r de la manière suivante :

$$B_E(r) = \{x \in E : \|x\| \leq r\}, \quad B_E^-(r) = \{x \in E : \|x\| < r\}.$$

Remarquons que $B_E(r)$ et $B_E^-(r)$ sont toutes deux des réseaux. Nous posons aussi $B_E(\infty) = B_E^-(\infty) = E$.

Cas des CDVF à précision finie

Dans certains cas, nous allons restreindre le cadre précédent au cas de ce que nous appelons des CDVF à précision finie. Nous les noterons ainsi. Soit \mathcal{K} un corps muni d'une valuation val . Nous demandons que \mathcal{K} soit complet par rapport à la norme définie par val . Nous notons $R = \mathcal{O}_{\mathcal{K}}$ son anneau des entiers, $m_{\mathcal{K}}$ son idéal maximal et $k_{\mathcal{K}} = \mathcal{O}_{\mathcal{K}}/m_{\mathcal{K}}$ son corps résiduel. Nous noterons CDVF (*complete discrete-valuation field*, corps complet discrètement valué) un tel corps. Nous renvoyons à Corps Locaux, de Serre [Ser79], pour une introduction à ces derniers. Soit $\pi \in R$ une uniformisante

de \mathcal{K} et soit $S_{\mathcal{K}} \subset R$ un système de représentants de $k_{\mathcal{K}} = O_{\mathcal{K}}/m_{\mathcal{K}}$ contenant 0. Tout élément de \mathcal{K} peut s'écrire de manière unique suivant un développement π -adique de la forme $\sum_{k \geq l} a_k \pi^k$, avec $l \in \mathbb{Z}$, et des $a_k \in S_{\mathcal{K}}$.

Plus particulièrement, le cas qui nous intéresse est celui où \mathcal{K} n'est pas nécessairement un corps effectif, mais où son corps résiduel $k_{\mathcal{K}}$ l'est (*i.e.* nous disposons d'algorithmes pour toutes les opérations de corps et pour tester l'égalité entre deux éléments de $k_{\mathcal{K}}$). Ainsi, des calculs formels, ou à précision infinie, peuvent être effectués sur des troncatures de développement π -adiques d'éléments de \mathcal{K} .³² Nous appelons *CDVF à précision finie* un tel corps, et *CDVR à précision finie* pour son anneau des entiers. Des exemples classiques de CDVF à précision finie sont $\mathcal{K} = \mathbb{Q}_p$, avec valuation p -adique, et $\mathbb{Q}[[X]]$ ou $\mathbb{F}_q[[X]]$ avec valuation X -adique. Nous supposons dans cette partie que \mathcal{K} est un CDVF à précision finie.

Dans un tel cas, les éléments de \mathcal{K} peuvent être manipulés symboliquement à une troncation près de leur développement π -adique. Ainsi, nous manipulerons des quantités de la forme $\sum_{i=k}^{d-1} a_i \pi^i + O(\pi^d)$, des approximations des nombres de \mathcal{K} , où $O(\pi^d)$ est une notation pour $\pi^d R$.

Définition 0.0.1. Pour étudier le comportement de la précision sur les approximations de nombres de \mathcal{K} , nous définissons l'**ordre** (ou la précision absolue) de $x = \sum_{i=k}^{d-1} a_i \pi^i + O(\pi^d)$ comme d . Nous définissons sa **précision relative** (ou nombre de chiffres significatifs) comme $d - \text{val}(\sum_{i=k}^{d-1} a_i \pi^i)$ si $\sum_{i=k}^{d-1} a_i \pi^i \neq 0$ et 0 sinon.

Modèle de complexité

Afin de rendre plus explicites les estimations que nous allons donner pour les algorithmes que nous étudierons, nous précisons la manière avec laquelle nous exprimons les temps de calcul.

Lorsque nous étudierons des algorithmes sur un corps dans le contexte de la précision infinie, les temps de calcul seront exprimés en nombre d'opérations sur le corps de base.

Lorsqu'il s'agira d'algorithmes sur un CDVF à précision finie K , nous compterons le nombre d'opérations sur des éléments de K avec une borne sur la précision.

La traduction d'une telle complexité en nombre d'opérations binaires dépend alors du corps considéré. Par exemple, il est bien connu (voir [VZGG13] Section 8.3), que multiplier deux entiers p -adiques connus à précision $O(p^N)$, ce qui se ramène essentiellement à la multiplication de deux entiers modulo p^N , est en pratique en $O(p^N \log p^N \log \log p^N)$ opérations binaires. De même, l'inversion d'un entier p -adique connu à précision $O(p^N)$ est en $O(p^N \log p^N \log \log p^N)$ opérations binaires. Le comportement est similaire pour ce qui est de la multiplication de séries formelles de $\mathbb{Z}/p\mathbb{Z}[[X]]$. Pour des résultats théoriques plus précis sur la complexité de ces opérations, nous renvoyons pour plus de détails à [Für09], [DKSS13] et finalement [HvdHL14b] et [HvdHL14a] qui obtiennent une complexité pour multiplier deux entiers de taille n en $O(n \log n 8^{\log^* n})$ et deux polynômes de degrés au plus n dans $\mathbb{Z}/p\mathbb{Z}[X]$ en $O(n \log n 8^{\log^* n} \log p)$. Si l'on s'intéresse à des séries formelles à coefficients dans un anneau où les opérations sont plus coûteuses, il faut bien sûr aussi tenir compte du coût des opérations dans cet anneau pour pouvoir estimer la complexité en nombre d'opérations binaires.

Contexte

Comme nous l'avons vu précédemment, l'algorithme p -adique a maintenant une longue histoire. Néanmoins, à chaque fois que l'on souhaite l'utiliser se pose naturellement deux questions : comment gérer la précision, et quelle manière de représenter les nombres p -adiques.

Nombres p -adiques, algorithmique et précision

Ne pouvant, par essence, pas être manipulés autrement qu'à précision finie, le problème de la gestion de la précision apparaît directement dès lors que l'on souhaite manipuler des nombres p -adiques.

^{32.} En pratique, dans le cas général, pour avoir des formules explicites pour ces opérations, il faut prendre pour $S_{\mathcal{K}}$ des représentants de Teichmüller et travailler avec des vecteurs de Witt.

Par exemple, dans l'algorithme de factorisation dans $\mathbb{Z}[X]$ de Berlekamp-Zassenhaus, pour factoriser un polynôme $f \in \mathbb{Z}[X]$, on débute par choisir un p adapté à notre polynôme. Un critère raisonnable est de prendre p premier avec le discriminant de f . Ensuite, on factorise $\bar{f} \in \mathbb{Z}/p\mathbb{Z}[X]$ et on remonte ses facteurs dans $\mathbb{Z}/p^n\mathbb{Z}[X]$ grâce à un lemme de Hensel. Dès que n est assez grand et quitte à recombinaison les facteurs obtenus, on en déduit une factorisation de f sur $\mathbb{Z}[X]$. Ainsi, le problème de la précision ici apparaît avec la question : quand est-ce que n est assez grand ? Une réponse *a priori* peut être donnée grâce à la borne de Mignotte, qui estime la taille des coefficients des facteurs de f en fonction des coefficients de f .

Un autre exemple est celui des algorithmes à la Kedlaya pour le comptage de points sur des courbes définies sur des corps finis. Ici, la réponse est obtenue grâce au calcul du polynôme caractéristique d'un endomorphisme Φ d'un espace de cohomologie adéquat de la courbe considéré donné par le morphisme de Frobenius. Cette espace de cohomologie est de dimension finie et à coefficients de nature p -adique. Pour estimer la précision nécessaire pour pouvoir lire le nombre de points recherché sur ce polynôme caractéristique, on utilise des estimés précis des valeurs propres de l'endomorphisme Φ donnés par la preuve de Deligne des conjectures de Weil. Là encore, une étude assez fine *a priori* est nécessaire.

Représentation classique et méthode directe de suivi de la précision

Classiquement, on représente les éléments de \mathbb{Q}_p en machine avec des approximations de la forme $p^k a + O(p^l)$ où $k, l \in \mathbb{Z}$ avec $l > k$ et $a \in \mathbb{N}$ premier avec p . $\mathbb{Z} \left[\frac{1}{p} \right]$ étant dense dans \mathbb{Q}_p , tout élément x de \mathbb{Q}_p s'approche bien par un élément de la forme $p^k a + O(p^l)$. On pourrait aussi dire que k et a s'obtiennent en tronquant le développement de Hensel de x . Une telle représentation est en quelque sorte l'analogue de celle choisie par l'arithmétique d'intervalles dans le cas réel.

Cette représentation étant choisie, il convient alors de s'intéresser au comportement de la précision, *i.e.* les $O(p^l)$, au cours de l'exécution d'un algorithme. Comme il existe des formules explicites pour connaître la précision sur le résultat d'une somme, différence, multiplication, ou division en fonction de ses entrées, il est possible d'appliquer ces formules les unes après les autres, ce que nous appelons la méthode directe de suivi de la précision. En particulier, la plus célèbre de ces formules est celle énonçant que

$$(p^k a + O(p^l)) + (p^{k'} b + O(p^l)) = (p^k a + p^{k'} b) + O(p^l),$$

dont est tiré le célèbre adage suivant : « les erreurs ne se cumulent pas en p -adique. »

Hélas, nous allons voir que les choses ne sont pas aussi idylliques dès que l'on s'intéresse à des calculs qui ne sont pas qu'une suite d'additions, différences, multiplications ou divisions, mais le calcul d'un objet plus complexe, ou encore dès que l'on s'intéresse à des calculs en dimension plus grande que 2.

Implémentations classiques p -adiques

La représentation précédente est celle qui est choisie par défaut en [S⁺11], sous le nom de "*capped relative precision*", et avec la contrainte supplémentaire d'avoir une borne sur la précision relative (le nombre de chiffres significatifs), sauf éventuellement pour 0, qui est connu à précision infinie.

Une autre possibilité en [S⁺11] est celle de "*capped absolute precision*", où l'on travaille avec une majoration sur les N pouvant apparaître dans un $O(p^N)$.

Enfin, une dernière possibilité en [S⁺11] est celle de "*fixed modulus*", où l'on travaille comme dans $\mathbb{Z}_p/p^N\mathbb{Z}_p$ mais avec la possibilité de faire des divisions sans tenir compte de la précision. Ce dernier choix peut amener des calculs plus rapides, mais sans garantie aucune sur la quantité de chiffres justes donnés sur le résultat.

Un autre logiciel ayant une implémentation importante des p -adiques est Magma [BCP97]. Dans celui-ci, deux modèles sont proposés pour travailler : *the fixed precision model* et *the free precision model*. Le premier est très similaire à la représentation "*fixed modulus*" de Sage, tandis que le second correspond plus à l'idée de la représentation "*capped relative precision*" de Sage, mais sans borne sur la précision relative.

Une dernière implémentation méritant d'être mentionnée est celle de Pari [BBB⁺13]. Celle-ci correspond essentiellement à la représentation "*capped relative precision*" de Sage.

Pour toutes ces implémentations, les opérations basiques sur les matrices, polynômes, ... sont disponibles³³. Néanmoins, le calcul n'est pas nécessairement optimisé du point de vue de la perte de précision.³⁴

Algorithmique détendue et en-ligne

Une représentation alternative des p -adiques a été développée récemment, en parallèle de la précédente. Il s'agit de l'approche de l'algorithmique détendue et en-ligne.

On peut définir un algorithme *en-ligne* sur les p -adiques comme un algorithme qui prenant en entrée des p -adiques, lit au plus les coefficients d'ordre n dans le développement de Hensel des entrées pour pouvoir écrire les coefficients d'ordre $n + 1$ de sa sortie. Cette notion est due à [Hen66]. Les algorithmes *détendus*, développés dans le sillage des travaux de Van der Hoeven publiés à la suite de [vdH97] et [vdH02] pour le cas des séries formelles, sont des algorithmes en-ligne qui, en un certain sens, minimisent le coût global du calcul. L'algorithmique détendue a été appliquée aux p -adiques dans [BvdHL11].

Ce nom d'algorithmique détendue a été choisi en opposition à deux autres types d'algorithmiques, les algorithmiques zélées et paresseuses. Sur les p -adiques, la première correspond aux calculs que l'on ferait directement selon la représentation classique présentée précédemment. La seconde, quant à elle, est une algorithmique en-ligne qui cherche à minimiser pour chaque n le nombre d'opérations à effectuer pour obtenir le n -ème coefficient de la sortie. Ceci peut occasionner une grande augmentation du temps de calcul entre ce qui est nécessaire pour calculer le n -ème coefficient et ce qui l'est pour le $(n + 1)$ -ème.

En opposition à ces méthodes, l'algorithmique détendue est en-ligne mais cherche, en faisant éventuellement quelques calculs supplémentaires par rapport au cas paresseux, à anticiper les calculs nécessaires pour obtenir le coefficient d'ordre $n + 1$ dès le calcul du coefficient d'ordre n . En conséquence, ceci entraîne un temps de calcul global, pour calculer tous les coefficients jusqu'à un N donné, moins important que dans l'algorithmique paresseuse. Nous renvoyons à [vdH02] pour plus de détails.

Dans le cas des p -adiques, l'algorithmique détendue s'applique particulièrement bien aux nombres p -adiques *récurifs*. Il s'agit essentiellement du sous-corps de \mathbb{Q}_p formé des coefficients des solutions d'équations de point fixe de la forme $Y = \Phi(Y)$ dans \mathbb{Q}_p^d , avec Φ algébrique, ou éventuellement rationnelle. Multiplication et division entre p -adiques récurifs peuvent être effectuées de manière détendue.

Une des grandes forces de l'algorithmique détendue sur les p -adiques récurifs est alors la suivante : une fois qu'un tel nombre est défini en machine, il vient avec un algorithme en-ligne donnant ses coefficients. En particulier, c'est l'utilisateur qui fixe le nombre le nombre de chiffres de précision qu'il souhaite obtenir, et l'algorithme les lui fournit.

En conséquence, ici, le problème de la gestion de la précision est laissé à l'implémentation et à la machine. Ceci peut être particulièrement confortable pour l'utilisateur qui n'a pas à se préoccuper d'estimer *a priori* la précision nécessaire pour que le calcul qui l'intéresse se passe correctement. Néanmoins, ceci ne dispense pas d'un suivi de la précision si l'on s'intéresse aux performances de son calcul. En effet, estimer la précision requise pour un algorithme est nécessaire pour pouvoir majorer le temps de calcul de cet algorithme.

33. En particulier, la forme normale de Smith l'est.

34. Notons aussi, pour la forme normale de Smith que seules des formes normales de Smith approchées sont, pour l'instant, renvoyées.

1. Méthode directe, applications et limites

“Thank you Mario! But our Princess
is in another castle!”

Super Mario Bros

Introduction

Dans ce chapitre, nous allons introduire et illustrer les méthodes classique de suivi direct de la précision sur des CDVF à précision finie. Pour cela, nous débutons, en Section 1.1 par présenter les formules de précision sur les opérations élémentaires, et illustrons les conséquences de l’absence d’erreur d’arrondi. Grâce à ces formules, nous illustrons en Section 1.2 la puissance du suivi direct de la précision sur l’exemple du calcul d’une forme échelonnée d’une matrice. Ceci nous servira dans le Chapitre 7 pour l’étude du calcul de bases de Gröbner sur des CDVF à précision finie. Ensuite, nous fournissons une nouvelle illustration de la puissance de cette méthode en Section 1.3 en étudiant le calcul de la forme normale de Smith sur des CDVF à précision finie. Ceci permet de donner une première solution au problème de la stabilité des systèmes linéaires, et nous l’appliquons au célèbre cas des matrices de Hilbert. Enfin, en Section 1.4, nous montrons les limites de ces méthodes directes à travers divers exemples, dont la suite récurrente SOMOS-4, qui nous accompagnerons tout au long de cette partie.

1.1. Suivi direct de la précision

1.1.1. Précision sur un CDVF à précision finie

Sur un CDVF à précision fini, il est assez aisé de suivre le comportement de la précision lorsqu’on effectue des opérations élémentaires comme l’addition ou la multiplication. En effet, soit $n_0 < m_0$, $n_1 < m_1$ des entiers, et $\varepsilon = \sum_{j=0}^{m_0-n_0-1} a_j \pi^j$, $\mu = \sum_{j=0}^{m_1-n_1-1} b_j \pi^j$, avec $a_j, b_j \in S_K$, et $a_0, b_0 \neq 0$. Il est bien connu que :

$$(\varepsilon \pi^{n_0} + O(\pi^{m_0})) + (\mu \pi^{n_1} + O(\pi^{m_1})) = \varepsilon \pi^{n_0} + \mu \pi^{n_1} + O(\pi^{\min(m_0, m_1)}).$$

En conséquence, l’addition de deux nombres connus à précision n est connue à précision n . Nous pouvons donner des formules aussi précises pour les autres opérations :

Proposition 1.1.1 (multiplication).

$$(\varepsilon \pi^{n_0} + O(\pi^{m_0}))(\mu \pi^{n_1} + O(\pi^{m_1})) = \varepsilon \mu \pi^{n_0+n_1} + O(\pi^{\min(m_0+n_1, m_1+n_0)}).$$

Proposition 1.1.2 (division).

$$\frac{\varepsilon \pi^{n_1} + O(\pi^{m_1})}{\mu \pi^{n_0} + O(\pi^{m_0})} = \varepsilon \mu^{-1} \pi^{n_1-n_0} + O(\pi^{\min(m_1-n_0, m_0+n_1-2n_0)}).$$

En conséquence, nous pouvons déjà voir pourquoi les CDVF à précision finie ont un comportement très différent de celui des nombres flottants : il n'y a pas d'erreur d'arrondi. Nous pourrions même être plus enthousiastes et énoncer que les erreurs ne s'additionnent pas. Nous verrons dans la suite de ce chapitre ce qu'il en est plus précisément.

Nous appellerons méthode directe de suivi de la précision le fait d'utiliser pas à pas les formules précédentes pour suivre la précision lors d'un calcul complexe ou pour étudier un algorithme.

1.1.2. Comparaison qualitative avec les réels

Les formules précédentes montrent bien qu'il y a déjà une différence de nature fondamentale entre la gestion de la précision sur les réels et sur les CDVF à précision finie : il n'y a pas d'erreurs d'arrondis lors de calculs sur des CDVF. Pour donner une idée des conséquences de cette différence, nous illustrons les effets des erreurs d'arrondis sur un exemple classique dans l'étude de la précision sur les réels.

Soit $f = 8 * x * (1 + x)^2 * (1 + 2x) * (x^2 + 3)^2 \in \mathbb{Q}[x]$ un polynôme. Nous pouvons calculer sa dérivée formellement et obtenir $f' = 128 * x^7 + 280 * x^6 + 768 * x^5 + 1240 * x^4 + 1344 * x^3 + 1224 * x^2 + 576 * x + 72$. Cependant, on peut souhaiter aussi estimer numériquement la valeur de cette dérivée en $x_0 = 0, 2 = 1/5$ par calcul d'un taux d'accroissement : $\frac{f(x_0+h)-f(x_0)}{h}$, pour h dont on espère que le prendre petit donnera une bonne estimation de $f'(x_0)$.¹

Nous définissons alors la fonction *Erreur* par $Erreur(h) = \left| \frac{f(x_0+h)-f(x_0)}{h} - f'(x_0) \right|$. Nous avons estimé, sur la figure 1.1 (page 36) le comportement d' $Erreur(h)$, en coordonnées logarithmiques, pour des nombres réels à précision relative 50 sur Sage. Nous remarquons que sur ce graphique, il y a alors trois phases. La première correspond aux valeurs de h comprises entre 10^{-20} et 10^{-3} , et $\log(Erreur(h))$ semble y croître linéairement avec $\log(h)$. Dans le vocabulaire de l'étude de la précision réelle, cela correspond au moment où les *erreurs de troncature* sont prédominantes, celles définies par h . La deuxième correspond à celles où h est entre environ 10^{-35} et 10^{-20} , et $\log(Erreur(h))$ semble y décroître linéairement avec $\log(h)$. Dans le vocabulaire de l'étude de la précision réelle, cela correspond au moment où les *erreurs d'arrondi* puis les erreurs dues au *nombre de chiffres significatifs* insuffisant sont prédominantes. La dernière phase, pour h inférieur à 10^{-35} correspond au moment où les chiffres significatifs sur h absorbent toute précision sur la sortie.

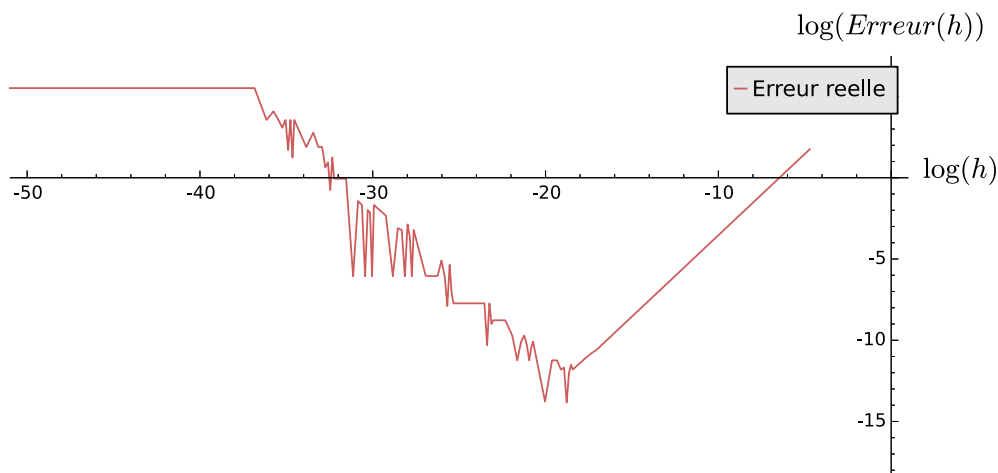


FIGURE 1.1. $-\log(Erreur(h))$ en fonction de $\log(h)$, pour $Erreur(h) = \left| \frac{f(x_0+h)-f(x_0)}{h} - f'(x_0) \right|$, sur \mathbb{R} à précision relative 50.

Nous pouvons tester le même calcul sur \mathbb{Q}_2 , avec là encore, précision relative $prec = 50$ sur \mathbb{Q}_2 en Sage. Nous avons estimé, sur la figure 1.2 (page 37) le comportement de $\log(|Erreur(h)|)$, en fonction de $\log|h|$. Nous remarquons que sur ce graphique, il n'y a cette fois que deux phases. La

1. Il existe bien sûr des méthodes plus efficaces et plus stables pour évaluer numériquement une dérivée, notamment en symétrisant le taux d'accroissement.

première est représentée pour $|h|$ entre 2^{-25} et 2^{-3} , et $\log(|\text{Erreur}(h)|)$ y croît linéairement avec $\log |h|$. Ceci correspond à nouveau au moment où les *erreurs de troncature* sont prédominantes, celles définies par h . La seconde est représentée pour $|h|$ entre 2^{-60} et 2^{-25} , et $\log(|\text{Erreur}(h)|)$ y décroît linéairement avec $\log |h|$. Ceci correspond exactement au moment où les chiffres significatifs sur h absorbent peu à peu la précision en sortie. Contrairement à la précision réelle, il n'y a pas de palier une fois que tous les chiffres significatifs sont absorbés par la précision sur h . Nous remarquons que le changement de pente sur la figure correspond exactement à la moitié de la précision. Ceci correspond exactement au fait qu'il n'y a pas d'erreurs d'arrondis. Nous remarquons aussi que pour une même précision relative $prec$, travailler sur \mathbb{Q}_2 fournit ici jusqu'à deux fois plus de chiffres significatifs que travailler sur \mathbb{R} . Nous remarquons enfin que faire varier p ne change ici rien de significatif numériquement à la courbe.

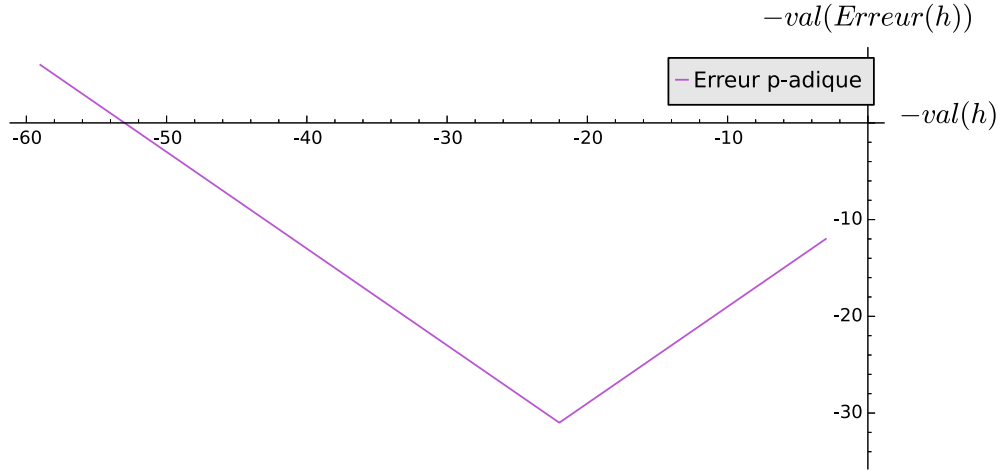


FIGURE 1.2. $-\log(|\text{Erreur}(h)|)$ en fonction de $\log |h|$, pour $\text{Erreur}(h) = \left| \frac{f(x_0+h)-f(x_0)}{h} - f'(x_0) \right|$, sur \mathbb{Q}_2 à précision relative 50.

Pour continuer l'étude qualitative, on peut vouloir comparer les deux courbes avec la Figure 1.3 (page 37).

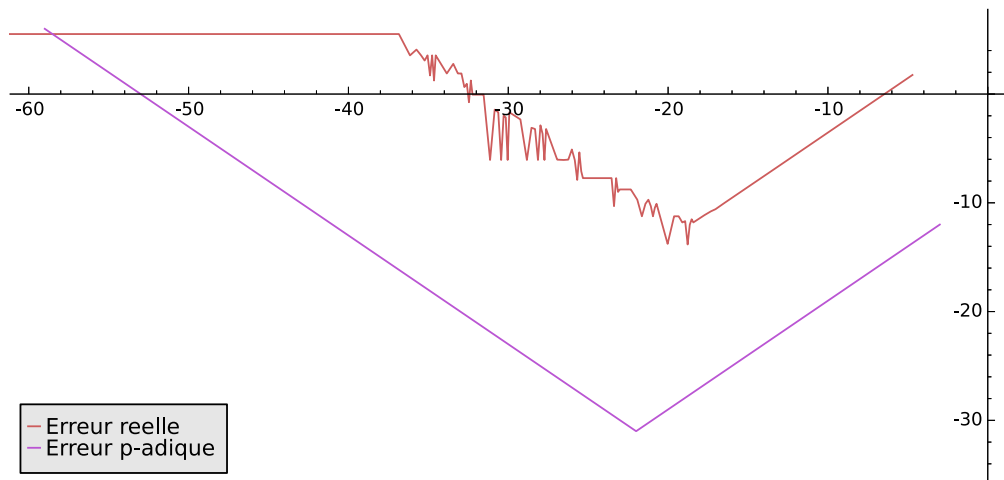


FIGURE 1.3. — Comparaison de $\log(|\text{Erreur}(h)|)$ en fonction de $\log |h|$, pour $\text{Erreur}(h) = \left| \frac{f(x_0+h)-f(x_0)}{h} - f'(x_0) \right|$, avec h pris dans \mathbb{Q}_2 à précision relative 50 ou dans \mathbb{R} à précision relative 50.

1.2. Application de cette méthode : le cas de l'échelonnement en lignes

Dans cette Section, nous appliquons l'étude classique de la perte de précision exposée précédemment, et plus précisément la Proposition 1.1.2 pour analyser un algorithme d'échelonnement en lignes par élimination gaussienne. Calculer un échelonnement en lignes est outil crucial en algèbre linéaire, et cette étude nous servira directement lors du Chapitre 7. Nous appliquerons cette méthode de la même manière pour obtenir une analyse de la précision lors du calcul d'une forme normale de Smith, ce qui permettra de donner une estimation suffisante de la perte de précision pour pouvoir résoudre des systèmes linéaires.

1.2.1. Algorithme d'échelonnement en lignes par élimination gaussienne

Nous commençons par expliciter ce que veut dire être sous forme échelonnée pour une matrice, et nous présentons l'algorithme d'échelonnement par élimination gaussienne que nous allons étudier.

Définition 1.2.1. Soit M une matrice de $M_{n,m}(\mathcal{K})$. Soit $ind_M : \{1, r\} \rightarrow \mathbb{Z}_{\geq 0} \cup \{\infty\}$ l'application qui envoie i sur l'indice de la colonne du premier coefficient non-nul de la ligne i de M . Alors M est dite sous forme échelonnée (en lignes) si ind_M est une application strictement croissante.

M est dite sous forme échelonnée à permutation près s'il existe une matrice de permutation $P \in M_n(\mathcal{K})$ telle que PM est sous forme échelonnée.

M' est une forme échelonnée de M si M' est sous forme échelonnée et s'il existe $P \in GL_n(\mathcal{K})$ telle que $M' = PM$. On définit la notion d'être une forme échelonnée à permutation près d'une matrice M de la même manière.

Exemple 1.2.2. La matrice dans $M_{4,5}(\mathbb{Q})$ suivante est sous forme échelonnée (en lignes) :

$$\begin{bmatrix} 1 & 3 & 4 & 3 & 2 \\ 0 & 0 & 2 & 4 & 3 \\ 0 & 0 & 0 & 1 & 7 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

L'algorithme suivant permet le calcul d'une forme échelonnée à permutation près d'une matrice à coefficients dans K valué :

Algorithme 1.2.3 : Échelonnement en lignes gaussien

entrée : \widetilde{M} , une matrice $n \times m$.

sortie : \widetilde{M} , une forme échelonnée (en lignes) de M à permutation près.

début

$\widetilde{M} \leftarrow M$;

si $n_{col} = 0$ ou $n_{row} = 1$ ou M n'a pas de coefficient non nul **alors**

Retourner \widetilde{M} ;

sinon

Trouver la ligne i du coefficient $M_{i,1}$ sur la première colonne qui a la valuation la plus petite ;

Permuter les lignes 1 et i ;

Par pivot avec la première ligne, éliminer les coefficients sur la première colonne des lignes qui ne sont pas la première ;

Procéder récursivement sur la matrice $\widetilde{M}_{i \geq 2, j \geq 2}$;

Retourner \widetilde{M} ;

Nous avons en effet la proposition suivante :

Proposition 1.2.4. Soit $M \in M_{n_l, n_c}(K)$ une matrice. Alors l'Algorithme 1.2.3 calcule une forme échelonnée à permutation près de M . Le temps de calcul est en $O(n_l n_c \times \min(n_l, n_c))$.

La propriété précédente est vraie lorsqu'on considère un corps sans prendre en compte la précision. Si nous travaillons sur un CDVF à précision finie \mathcal{K} , nous insistons sur le fait que lors de l'élimination par pivot, nous devons mettre des vrai zéros sur la colonne sur laquelle nous pivotons, et non des $O(\pi^k)$. Dans le cas contraire, la matrice en sortie n'est pas sous forme échelonnée (à permutation près) mais seulement sous forme échelonnée approchée. La Sous-Section suivante explique comment, sous certaines hypothèses, nous pouvons procéder pour pivoter de façon à ce que l'algorithme précédent renvoie bien une forme échelonnée, même sur une matrice dont les coefficients sont connus à précision finie.

1.2.2. Comment pivoter

Nous étudions ici comment éliminer des coefficients sur une colonne avec le pivot, et l'effet que cette action a sur la précision.

Proposition 1.2.5. *Soit $n_0 \leq n_1 < n$ des entiers, et $\varepsilon = \sum_{j=0}^{n-n_1-1} a_j \pi^j$, $\mu = \sum_{j=0}^{n-n_0-1} b_j \pi^j$, avec $a_j, b_j \in S_{\mathcal{K}}$, et $a_0, b_0 \neq 0$.*

Pour mettre un vrai zéro sur le coefficient $M_{i,j} = \varepsilon \pi^{n_1} + O(\pi^n)$, lors de l'exécution de l'Algorithme 1.2.3, nous l'éliminons avec le pivot $piv = M_{(i,i)} \mu \pi^{n_0} + O(\pi^n)$ sur la ligne L . Ceci est effectué par l'opération suivante sur la i -ème ligne L_i :

$$L_i \leftarrow L_i - \frac{M_{i,j}}{piv} L = L_i + (\varepsilon \mu^{-1} \pi^{n_1-n_0} + O(\pi^{n-n_0})) L,$$

avec en même temps l'opération symbolique $M_{i,j} \leftarrow 0$.

Démonstration. L'opération symbolique $M_{i,j} \leftarrow 0$ est simplement une partie de l'opération symbolique $L_i \leftarrow L_i - \frac{M_{i,j}}{piv} L$. Cependant, pour tout autre coefficient de L_i , nous ne connaissons pas le résultat de l'opération avec précision infinie, et à précision finie, ce qui est fait est $L_i + (\varepsilon \mu^{-1} \pi^{n_1-n_0} + O(\pi^{n-n_0})) L$.

En effet, nous montrons que $\frac{M_{i,j}}{piv} = \varepsilon \mu^{-1} \pi^{n_1-n_0} + O(\pi^{n-n_0})$.

C'est une conséquence directe de la Proposition 1.1.2 : $\frac{M_{i,j}}{piv} = \frac{\varepsilon \pi^{n_1} + O(\pi^n)}{\mu \pi^{n_0} + O(\pi^n)}$, et ainsi $\frac{M_{i,j}}{piv} = \varepsilon \mu^{-1} \pi^{n_1-n_0} + O(\pi^{n-n_0})$, puisque $\min(n + n_1 - 2n_0, n - n_0, n + n - 2n_0) = n - n_0$. \square

1.2.3. Calcul de la forme échelonnée

Nous sommes maintenant capables de suivre la perte de précision lorsque l'on utilise l'Algorithme 1.2.3 pour calculer une forme échelonnée en lignes d'une matrice. Le résultat est le suivant :

Théorème 1.2.6. *Soit M une matrice $n \times m$ ($0 \leq n \leq m$) avec coefficients dans R tous connus à la précision $k \geq 0$ et tels que le mineur principal $\Delta = \det((M_{i,j})_{1 \leq i \leq n, 1 \leq j \leq n})$ satisfait $\text{val}(\Delta) < k$.*

Alors, la perte de précision maximale lors de calcul de l'échelonnement en lignes de M est majorée par $\text{val}(\Delta)$.

Démonstration. Pour prouver ce résultat, nous étudions d'abord le procédé d'élimination par pivot, et nous concluons par récurrence sur le nombre de lignes.

Lorsqu'on calcule l'échelonnement matriciel par l'Algorithme 1.2.3, la première étape est de chercher le coefficient sur la première colonne $M_{i,1}$ avec plus petite valuation. Ensuite, par permutation de lignes, nous le plaçons sur la première ligne, L_1 . Appelons piv ce coefficient et n_1 sa valuation.

Comme dans la Proposition 1.2.5, nous éliminons ensuite les coefficients sur les lignes L_i ($i \neq 1$) par l'opération suivante $L_i \leftarrow L_i - \frac{M_{i,j}}{piv} L_1$. Appelons $M^{(1)}$ la matrice que l'on obtient ainsi. Alors, les coefficients de la sous-matrice $M_{2 \leq i \leq n, 2 \leq j \leq m}^{(1)}$ sont connus à précision $O(\pi^{n-n_1})$.

Nous pouvons alors précéder récursivement sur la sous-matrice $M_{2 \leq i \leq n, 2 \leq j \leq m}^{(1)}$.

Le résultat que nous souhaitons montrer est clair pour une matrice avec $n = 1$ lignes. Nous pouvons aussi remarquer que dans l'échelonnement de la première colonne, les opérations effectuées sur les lignes ne change la valeur du mineur principal que d'un signe. Nous avons $\Delta = \pm piv \times \det M_{2 \leq i \leq n, 2 \leq j \leq n}^{(1)}$. Ceci ne modifie pas sa valuation.

En conséquence, le résultat est clair par récurrence sur le nombre de colonnes. \square

1.2.4. Un résultat plus précis

Le résultat suivant, utilisant de manière cruciale le fait d'être sur un CDVF, permet d'estimer la perte de précision lorsqu'on est de rang plein. De plus, il nous montre l'optimalité, en un certain sens, du choix des pivots effectué par l'Algorithme 1.2.3

Proposition 1.2.7. *Soit M une matrice $n \times m$ ($0 \leq n, m$) avec coefficients dans R tous connus avec la même précision $k \geq 0$. Soit $l \leq m$ tel qu'il existe un l -mineur sur les l -premières colonnes C_1, \dots, C_l , avec valuation strictement inférieure à k .*

Soit Δ le produit des pivots obtenus lors du calcul de la forme échelonnée de M par l'Algorithme 1.2.3 jusqu'à la colonne l .

Alors, la perte de précision maximale observée lors du calcul de la forme échelonnée (en lignes) de M jusqu'à colonne l , est majorée par $\text{val}(\Delta)$ et de plus, $\text{val}(\Delta)$ est la plus petite valuation d'un l -mineur sur les l premières colonnes de M .

Démonstration. Ceci vient du fait suivant : dans l'anneau des entiers d'un corps complet de valuation discrète, un idéal I est engendré par n'importe lequel de ses éléments qui atteint $\min(\text{val}(I))$.

Notons (C_1, \dots, C_l) les l premières colonnes de M . Nous définissons $I_{l\text{-mineurs}}$ comme l'idéal de R engendré par les l -mineurs sur (C_1, \dots, C_l) , alors $I_{l\text{-mineurs}}$ reste inchangé par les opérations sur les lignes effectuées lors du calcul de la forme échelonnée par élimination de Gauss. En effet, ces opérations correspondent à multiplier à gauche par une matrice inversible, ce qui conserve clairement $I_{l\text{-mineurs}}$.

Une fois que le calcul de la forme échelonnée jusqu'à la l -ème colonne est fini, il ne reste qu'un seul mineur non-nul sur les l -premières colonnes. Il correspond à une matrice triangulaire supérieure dont les coefficients diagonaux sont les pivots utilisés lors de l'échelonnement. La valeur de ce mineur est ainsi Δ et on a $I_{l\text{-mineurs}} = \langle \Delta \rangle$.

En conclusion, Δ engendre I_{mineur} et sa valuation atteint donc $\min(\text{val}(I))$. Ceci conclut la preuve. \square

En conséquence, le calcul d'une forme échelonnée par élimination gaussienne sur une matrice M jusqu'à la colonne l fournit un choix de pivot qui engendre la plus petite perte de précision que l'on pourrait obtenir par échelonnement.

Remarque 1.2.8. Nous pourrions étendre l'étude précédente au calcul d'une forme échelonnée réduite (i.e. où il n'y a qu'un coefficient non nul par colonne où l'on prend un pivot). Cela sera fait, dans un contexte tropical au Chapitre 9.

1.3. Résolution de systèmes linéaires et forme normale de Smith

Le but de cette Section est d'étudier par des méthodes classiques quelle est la perte de précision lors de la résolution d'un système matriciel $Y = AX$ où A est une matrice inversible. Nous allons voir qu'avec un calcul adapté de la forme normale de Smith de A , on peut obtenir une perte de précision bien moindre que ce que l'on obtiendrait avec un échelonnement réduit. L'étude de formes normales de Smith (approchées) sur des corps valués n'est pas nouvelle. Elle est déjà présente dans [DSV01] pour le calcul de formes normales de Smith sur \mathbb{Z} via un calcul dans le localisé $\mathbb{Z}_{(p)}$. Elle l'est aussi dans [AKR09] pour le calcul d'une majoration du co-rang d'une matrice dont les coefficients sont connus à précision $O(\pi^m)$

1.3.1. Forme normale de Smith

Nous commençons par donner une définition, sous forme de proposition, de la forme normale de Smith d'une matrice dans $M_{n,m}(\mathcal{K})$:

Proposition 1.3.1. *Soit $M \in M_{n,m}(\mathcal{K})$ une matrice. Il existe $P \in GL_n(O_{\mathcal{K}})$, $\det P = \pm 1$,*

$Q \in GL_m(O_K)$, $\det Q = \pm 1$ et $\Delta \in M_{n,m}(K)$ de la forme

$$\Delta = \begin{bmatrix} \pi^{a_1} & & & & \\ & \ddots & & & \\ & & \pi^{a_s} & & \\ & & & 0 & \\ & & & & 0 \end{bmatrix},$$

avec $a_1 \leq \dots \leq a_s$, et tels que $M = P\Delta Q$.

Δ est unique, et est appelé la forme normale de Smith de M , et on dit que P, Δ, Q réalisent la forme normale de Smith de M . Δ est un invariant des orbites de $M_{n,m}(K)$ sous l'action de $GL_n(O_K) \times GL_m(O_K)$ donnée par $(A, B) \cdot M = AMB^{-1}$, et les a_i sont appelés les facteurs invariants de M . Nous noterons $\sigma_i(M) = \text{val}(a_i)$ pour tout $i \in \llbracket 1, \min(n, m) \rrbracket$.

Remarque 1.3.2. Par rapport à la forme normale de Smith définie classiquement sur un anneau principal, la principale différence consiste en le choix *a priori* d'une forme canonique pour les facteurs invariants, définie par le choix d'une uniformisante.

1.3.2. Calcul de la forme normale de Smith

Dans cette Sous-Section, nous allons donner puis étudier un algorithme pour calculer une réalisation d'une forme normale de Smith. Nous allons montrer que, sous certaines hypothèses, il est possible de calculer la forme normale de Smith d'une matrice dont les coefficients dans R sont connus à précision finie. Cependant, contrairement au cas du calcul d'une forme échelonnée par pivot de Gauss, nous n'allons pas procéder directement pour calculer la forme normale de Smith, mais plutôt calculer d'abord une forme approchée, et ensuite la forme normale de Smith. Ceci permettra de ne pas accumuler les erreurs dues aux divisions.

Matrices des opérations élémentaires

Afin de pouvoir exprimer explicitement l'algorithme de calcul de la forme normale de Smith d'une matrice, nous définissons les matrices des opérations élémentaires sur les lignes et les colonnes que nous effectuerons.

Définition 1.3.3. Nous définissons la matrice $E_{i,j,n}$ pour $n \in \mathbb{N}^*$ et $i, j \leq n$ comme la matrice de $M_n(K)$ dont le coefficient d'ordre (u, v) est $\delta_{u,i}\delta_{v,j}$. Nous pouvons alors définir les matrices des opérations élémentaires, pour i, j, n comme précédemment :

- Permutation : $\text{Perm}_n(i, j) = Id_n + E_{i,j,n} + E_{j,i,n} - E_{i,i,n} - E_{j,j,n}$.
- Transvection : pour $x \in K$, $\text{Transvec}_n(i, j, x) = Id_n + xE_{j,i,n}$.
- Dilatation : pour $x \in K^\times$, $\text{Dilat}_n(i, x) = Id_n + xE_{i,i,n} - E_{i,i,n}$.

L'effet sur les lignes d'une matrice M de la multiplication XM où X est l'une de ces matrices est bien connu. De même pour l'effet sur les colonnes de M pour MX .

Calcul d'une forme normale de Smith approchée

Pour ce faire, nous définissons d'abord ce que nous appelons une forme normale de Smith approchée.

Définition 1.3.4. Soit $M \in M_{n,m}(K)$, connue à précision $O(\pi^l)$. Nous appelons **décomposition de Smith approchée** de M une factorisation

$$M = U\Delta V$$

avec $U \in M_n(R)$, $V \in M_m(R)$ de déterminant ± 1 connus à précision $O(\pi^l)$ et $\Delta \in M_{n,m}(K)$ tel que $\Delta = \Delta_0 + O(\pi^l)$, où $\Delta_0 \in M_{n,m}(K)$ dont les seuls coefficients non-nuls sont diagonaux. Nous demandons de plus que les coefficients diagonaux de Δ_0 soit $\Delta_0[1, 1] = \pi^{\alpha_1}, \dots, \Delta_0[\min(n, m), \min(n, m)] = \pi^{\alpha_{\min(n, m)}}$ avec $\alpha_1 \leq \dots \leq \alpha_{\min(n, m)}$. $\alpha_i = +\infty$ est possible.

1. Méthode directe, applications et limites

Pour calculer une forme normale de Smith approchée, nous pouvons utiliser l'algorithme suivant :

Algorithme 1.3.5 : SNFApprochee : Calcul de forme normale de Smith approchée

entrée : M , une matrice $n \times m$, connue à précision $O(\pi^l)$, $P \in M_n(R)$ et $Q \in M_m(R)$.

sortie : U, Δ, Q avec (U, Δ, V) , connus à précision $O(\pi^l)$, réalisant une forme normale de Smith approchée de M , $M = U\Delta V$

début

```

 $\widetilde{M} \leftarrow M$  ;
si Si  $M$  est vide ou tous les coefficients de  $M$  sont des  $O(\pi^l)$  alors
    | Retourner  $\widetilde{M}$  ;
sinon
    Trouver  $i, j$  tel que le coefficient  $M_{i,j}$  atteigne  $\min_{k,l} \text{val}(M_{k,l})$  (si égalité, prendre le
    plus petit indice pour l'ordre lexicographique) ;
    Écrire  $a\pi^\alpha + O(\pi^l) = M_{i,j}$  (avec  $\text{val}(M_{i,j}) = \alpha$ ) ;
     $\widetilde{M} := \text{Perm}_n(1, j) \cdot \widetilde{M}$  ;
     $P := P \cdot \text{Perm}_n(1, j)$  ;
     $\widetilde{M} := \widetilde{M} \cdot \text{Perm}_m(1, j)$  ;
     $Q := \text{Perm}_m(1, j) \cdot Q$  ;
     $\widetilde{M} := \text{Dilat}_n(1, a^{-1}) \cdot \widetilde{M}$  ;
     $P := P \cdot \text{Dilat}_n(1, a)$  ;
    pour  $i$  de 2 à  $n$  faire
        | Écrire  $b\pi^k = \widetilde{M}_{i,1}$  ;
        |  $\widetilde{M} := \text{Transvec}_n(1, i, b\pi^{k-\alpha}) \cdot \widetilde{M}$  ;
        |  $P := P \cdot \text{Transvec}_n(1, i, -b\pi^{k-\alpha})$  ;
    pour  $j$  de 2 à  $m$  faire
        | Écrire  $b\pi^k = \widetilde{M}_{1,j}$  ;
        |  $\widetilde{M} := \widetilde{M} \cdot \text{Transvec}_n(j, 1, a^{-1}b\pi^{k-\alpha})$  ;
        |  $Q := \text{Transvec}_n(j, 1, -a^{-1}b\pi^{k-\alpha}) \cdot Q$  ;
    Récursivement :  $(P', \Delta', Q') := \text{SNFApprochee}(\widetilde{M}_{i \geq 2, j \geq 2}, Id_{n-1}, Id_{m-1})$  ;
    Augmenter  $P'$  en une matrice  $n \times n$  ;
    Augmenter  $Q'$  en une matrice  $m \times m$  ;
    Augmenter  $\Delta'$  en une matrice  $\Delta$  de taille  $n \times m$  avec  $\Delta[1, 1] = \pi^\alpha + O(\pi^l)$  ;
    Retourner  $P \cdot P', \Delta, Q' \cdot Q$  ;

```

Remarque 1.3.6. Soit $A \in M_n(\mathcal{K})$ une matrice. Nous disons que nous **augmentons** A en $A' \in M_{n+1, n+1}(\mathcal{K})$ en posant :

$$A' = \begin{bmatrix} 1 & 0 & & 0 \\ 0 & & & \\ \vdots & & \boxed{A} & \\ 0 & & & \end{bmatrix}.$$

C'est ce qui est utilisé dans la dernière partie de l'Algorithme 1.3.5.

Nous avons alors la proposition suivante :

Proposition 1.3.7. Prenant en entrée M , une matrice $n \times m$, connue à précision $(O(\pi^l)$ et $P = Id_n$ et $Q = Id_m$, l'algorithme SNFApprochee termine et renvoie U, Δ, V réalisant une forme normale de Smith approchée de M . U, Δ et V sont connus à précision $O(\pi^l)$. Le temps de calcul est en $O(\min(n, m) \max(n, m)^2)$.

Démonstration. Contrairement au cas du calcul de la forme normale de Smith sur un anneau euclidien, le fait que l'on travaille sur un corps discrètement valué amène qu'après la phase d'élimination (approchée) par pivot sur la première colonne et la première ligne, tous les coefficients de $\widetilde{M}_{i \geq 2, j \geq 2}$ sont de valuation plus grande que $\text{val}(\widetilde{M}[1, 1])$. Ceci explique que contrairement au

cas où l'on veut calculer la forme normale de Smith sur un anneau euclidien, il n'est pas nécessaire après élimination de chercher à nouveau un pivot sur tout \widetilde{M} qui atteint la plus petite valuation. Le fait que l'algorithme termine et calcule bien une forme normale de Smith approchée est clair. Le seul point à vérifier est que les opérations de transvections mettent bien des coefficients $O(\pi^l)$ lorsqu'elles éliminent un coefficient, et que les matrices P et Q sont toujours connues à précision $O(\pi^l)$, mais ceci est direct avec la Proposition 1.1.1. \square

Déduction de la forme normale de Smith

Nous montrons ici comment déduire la forme normale de Smith d'une forme normale de Smith approchée, et estimons la perte de précision totale pour le calcul de cette forme normale. Pour cela, nous utilisons l'algorithme suivant :

Algorithme 1.3.8 : SNFPrecisee : Passage d'une forme approchée à la forme normale de Smith

entrée : M , une matrice $n \times m$ de rang plein, connue à précision $O(\pi^l)$, et (U, Δ, V) , connus à précision $O(\pi^l)$, réalisant une forme normale de Smith approchée de M , $M = U\Delta V$.
On suppose que pour tout i , $\text{val}(\Delta[i, i]) < l$.
sortie : Δ_0 une matrice diagonale tel que $\text{val}(\Delta_0[1, 1]) \leq \dots \leq \text{val}(\Delta_0(\min(n, m), \min(n, m)))$, avec $k = \max_i \text{val}(\Delta_{i,i})$, et U', V' connus à précision $O(\pi^{l-k})$ tels que $M = U'\Delta_0V'$.

début

```

 $\Delta_0 \leftarrow \Delta$  ;
 $t := \min(n, m)$  ;
pour  $i$  de 1 à  $t$  faire
    Écrire  $\pi^{a_i} + O(\pi^l) := \Delta_0[i, i]$   $\Delta_0 := \text{Dilat}_n(i, \pi^{a_i}/\Delta_0[i, i]) \cdot \Delta_0$ ;
     $P := P \cdot \text{Dilat}_n(1, \pi^{a_i}/\Delta_0[i, i])$ ;
    si  $t = m$  alors
        pour  $j$  de 1 à  $n$ ,  $j \neq i$  faire
             $\Delta_0 := \text{Transvec}_n(j, i, \Delta_0[j, i]/\pi^{a_i}) \cdot \widetilde{M}$ ;
             $P := P \cdot \text{Transvec}_n(j, i, \Delta_0[j, i]/\pi^{a_i})$ ;
        sinon
            pour  $j$  de 1 à  $m$ ,  $j \neq i$  faire
                 $\Delta_0 := \widetilde{M} \cdot \text{Transvec}_n(i, j, \Delta_0[i, j]/\pi^{a_i})$ ;
                 $Q := \text{Transvec}_n(i, j, \Delta_0[i, j]/\pi^{a_i}) \cdot Q$ ;
    Retourner  $\Delta_0, P, Q$  ;
    
```

Proposition 1.3.9. *Étant donnés M , une matrice $n \times m$ de rang plein, connue à précision $O(\pi^l)$, et (U, Δ, V) , connus à précision $O(\pi^l)$, réalisant une forme normale de Smith approchée de M , $M = U\Delta V$, avec pour tout i , $\text{val}(\Delta[i, i]) < l$, l'Algorithme 1.3.8 calcule U', Δ_0, V' avec $M = U'\Delta_0V'$, Δ_0 la forme normale de Smith de M . De plus, si b est la plus grande valuation des facteurs invariants de M , alors U' et V' sont connus à précision $O(\pi^{l-b})$. Le temps de calcul est en $O(\max(n, m)^2)$.*

Démonstration. La preuve est tout à fait similaire à celle de la Proposition 1.2.5 : les dilatations sont sans effet sur la précision (avec la proposition 1.1.2) et concernant les transvections, celles sur Δ_0 éliminent (de manière symbolique) les coefficients non-diagonaux, et sont vus comme ajouter des $O(\pi^{l-a_i})$ hors du coefficient à éliminer ou lorsqu'elles sont effectuées sur P et Q . Le résultat est alors clair. Pour ce qui est du temps de calcul, nous pouvons remarquer que tout l'algorithme peut se résumer à modifier la précision sur coefficients des matrices P, Q et Δ , d'où le résultat. \square

En conclusion, nous pouvons énoncer le résultat suivant concernant le calcul de la forme normale de Smith (d'une matrice de rang plein) :

Théorème 1.3.10. *Soit M une matrice $n \times m$ de rang plein, connue à précision $O(\pi^l)$. Soit b la plus grande valuation des facteurs invariants de M . Supposons que $l > b$. Alors, en appliquant les Algorithmes 1.3.5 puis 1.3.8, nous calculons P, Q, Δ avec $M = P\Delta Q$ et Δ forme normale de Smith*

1. Méthode directe, applications et limites

de M . De plus, les coefficients de P et Q sont connus à précision $O(\pi^{l-b})$. Le temps de calcul est en $O(\max(n, m)^2 \min(n, m))$.

Proposition 1.3.11. *Dans le contexte du théorème précédent, soit P et Q les matrices fournies par les Algorithmes 1.3.5 puis 1.3.8 tels que $M = P\Delta Q$. Alors on peut modifier les Algorithmes 1.3.5 et 1.3.8 pour obtenir les inverses de P et Q à précision $O(\pi^{l-b})$. En conséquence, si $M \in M_n(\mathcal{K})$ est inversible, nous pouvons déduire l'inverse M^{-1} de M en écrivant :*

$$M^{-1} = Q^{-1}\Delta^{-1}P^{-1}.$$

La perte (ou le gain) de précision est donné par la plus grande valuation d'un facteur invariant de M .

Démonstration. Il suffit de faire les calculs des inverses de P et Q itérativement. Cela peut se faire en effectuant à chaque multiplication du P en construction par une matrice de dilatation, transvection ou permutation, la multiplication du P^{-1} en construction par l'inverse de la matrice correspondante. De même pour Q . \square

Remarque 1.3.12. D'un point de vue de la précision, la Proposition précédente est plus efficace pour $M \in M_n(O_{\mathcal{K}}) \cap GL_n(\mathcal{K})$ qu'appliquer naïvement la formule $M^{-1} = \frac{1}{\det A} {}^t \text{com}(A)$. En effet, si $\Delta = \text{Diag}(\pi^{a_1}, \dots, \pi^{a_n})$ est la forme normale de Smith de M , $\text{val}(\det a) = \sum_{i=1}^n a_i \geq a_n$.

Un exemple

Exemple 1.3.13. Soit $M = \begin{bmatrix} 2 + O(2^6) & 4 + O(2^6) & 4 + O(2^6) & 6 + O(2^6) \\ -6 + O(2^6) & 6 + O(2^6) & 12 + O(2^6) & O(2^6) \\ 10 + O(2^6) & -4 + O(2^6) & -16 + O(2^6) & 6 + O(2^6) \end{bmatrix}$ une matrice de $M_{3,4}(\mathbb{Q}_2)$ dont les coefficients sont connus à précision $O(2^6)$.² Après application de l'Algorithme

d'échelonnement approché 1.3.5, nous obtenons : $P = \begin{bmatrix} 1 & 0 & 0 \\ 1/3 & 1/9 & 0 \\ 1 & -4/3 & -1 \end{bmatrix}$, $Q = \begin{bmatrix} 1 & -2 & 2/3 & -1 \\ 0 & 1 & -4/3 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

et

$$\Delta = \begin{bmatrix} 2 + O(2^6) & O(2^6) & O(2^6) & O(2^6) \\ O(2^6) & 2 + O(2^6) & O(2^6) & O(2^6) \\ O(2^6) & O(2^6) & 4 + O(2^6) & O(2^6) \end{bmatrix}$$

tels que $PMQ = \Delta$.

Nous appliquons alors l'Algorithme de calcul de forme normale de Smith approchée pour

obtenir $P = \begin{bmatrix} 1 + O(2^5) & 0 & 0 \\ 1/3 + O(2^5) & 1/9 + O(2^5) & 0 \\ 1 + O(2^4) & -4/3 + O(2^4) & -1 + O(2^4) \end{bmatrix}$, $Q = \begin{bmatrix} 1 & -2 & 2/3 & -1 \\ 0 & 1 & -4/3 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ et $\Delta =$

$\begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 4 & 0 \end{bmatrix}$ tels que $PMQ = \Delta$. Ainsi, nous avons bien calculé la forme normale de Smith de

M ainsi qu'une réalisation de celle-ci.

Remarquons qu'il est possible de mener le calcul d'une forme normale de Smith approchée directement en Sage, Magma ou Pari. Par contre, à notre connaissance en 2015, la déduction d'une forme normale de Smith exacte à partir d'une forme normale de Smith approchée n'y est pas implémentée. Nous renvoyons les lecteurs intéressés pour une implémentation en Sage à la page suivante : <http://perso.univ-rennes1.fr/tristan.vaccon/fglm.sage>, où une version (non optimisée) est disponible.

². Même si nous travaillons ici dans \mathbb{Q}_2 , nous avons choisi de garder dans nos exemples l'écriture décimale ou fractionnaire pour des raisons de place.

1.3.3. Résolution de systèmes linéaires.

Grâce aux résultats de la Sous-Section précédente, nous pouvons estimer quelle est la perte de précision lors de la résolution d'un système linéaire :

Théorème 1.3.14. *Soit $M \in M_n(\mathcal{K})$ une matrice inversible. Soit $M = P\Delta Q$ la forme normale de Smith de M . Soit $l \in \mathbb{N}^*$ tel que $l > b = \max_{1 \leq i \leq n} \text{val}(\Delta[i, i])$. Supposons que les coefficients de M soient connus à précision $O(\pi^l)$. Soit $Y \in \mathcal{K}^n$ connu à précision $O(\pi^l)$. Alors, on peut résoudre $Y = MX$ et X est connu à précision $O(\pi^{l-2b})$.*

Démonstration. Pour résoudre $Y = MX$, il suffit d'avoir $Y = P\Delta QX$. Or, d'après la proposition 1.3.11, on connaît P^{-1} et Q^{-1} à précision $O(\pi^{l-b})$ et Δ est connu à précision infinie, donc Δ^{-1} aussi. On a alors $X = Q^{-1}\Delta^{-1}P^{-1}Y$ où Y connu à précision $O(\pi^l)$, P^{-1} , Q^{-1} connus à précision $O(\pi^{l-b})$ et Δ^{-1} qui amène une multiplication par π^{-b} de la dernière ligne de $P^{-1}Y$. En conclusion, avec la Proposition 1.1.1, X est connu à précision $O(\pi^{l-2b})$. \square

Lorsque M n'est pas carré, mais que nous pouvons assurer que $Y \in \text{Im}(M)$, alors nous avons la variante suivante :

Proposition 1.3.15. *Soit $M \in M_{n,m}(\mathcal{K})$ une matrice de rang plein. Soit $M = P\Delta Q$ la forme normale de Smith de M . Soit $l \in \mathbb{N}^*$ tel que $l > b = \max_{1 \leq i \leq \min(n,m)} \text{val}(\Delta[i, i])$. Supposons que les coefficients de M soient connus à précision $O(\pi^l)$. Soit $Y \in \mathcal{K}^n$ connu à précision $O(\pi^l)$ tel que $Y \in \text{Im}(M)$. Alors, on peut obtenir X , connu à précision $O(\pi^{l-2b})$, tel que $Y = MX$. Si $Y \in \mathcal{R}^n$, les coefficients de X sont de valuation au moins $-b$.*

Démonstration. L'idée de démonstration est essentiellement la même que pour le Théorème 1.3.14, mais cette fois-ci Δ n'est pas inversible. En conséquence, remarquons que si $Y \in \text{Im}(M)$, alors $P^{-1}Y \in \text{Im}(\Delta)$. Ceci implique que les coefficients de $P^{-1}Y$ d'indice strictement plus grand que $\min(n, m)$ sont nuls. Il est alors possible d'écrire $P^{-1}Y = \Delta X_0$ avec X_0 connu à précision $O(\pi^{l-2b})$, $P^{-1}Y$ étant connu à précision $O(\pi^{l-b})$. Si $Y \in \mathcal{R}^n$, les coefficients de X_0 sont de valuation au moins $-b$. Finalement, $X = Q^{-1}X_0$ est connu à précision $O(\pi^{l-2b})$ et convient. \square

1.3.4. Application aux matrices de Hilbert

Nous présentons ici une application de la méthode de calcul d'inverse développée en Proposition 1.3.11 à l'étude d'une famille de matrices particulière, les matrices de Hilbert.

Présentation de l'étude des matrices de Hilbert

Nous commençons par donner une définition :

Définition 1.3.16. Soit $n \in \mathbb{N}^*$, nous définissons la matrice de Hilbert de dimension n , $H^{(n)} \in M_n(\mathbb{Q})$, par $H_{i,j}^{(n)} = \frac{1}{i+j-1}$ pour tout $1 \leq i, j \leq n$.

Par exemple, nous avons :

$$H^{(5)} = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} \\ \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} & \frac{1}{8} \\ \frac{1}{5} & \frac{1}{6} & \frac{1}{7} & \frac{1}{8} & \frac{1}{9} \end{bmatrix}.$$

Ces matrices ont été introduites par David Hilbert pour l'étude de problèmes d'approximation, au sens des moindres carrés, de fonctions par des polynômes. Celles-ci ont par ailleurs, l'intérêt de fournir un exemple complètement explicite de matrice dont le conditionnement sur \mathbb{R} pour la norme subordonnée à la norme $\|\cdot\|_2$ est grand, et croît exponentiellement avec la dimension n (voir [Tod54]).

1. Méthode directe, applications et limites

Leur étude est cependant aisée grâce au fait qu'il s'agit de matrices de Cauchy. En particulier, l'inverse de $H^{(n)}$, que nous noterons $H^{(n)-1}$ est donnée explicitement par :

$$H_{i,j}^{(n)-1} = (-1)^{i+j} (i+j-1) \binom{n+i-1}{n-j} \binom{n+j-1}{n-i} \binom{i+j-2}{i-1}^2,$$

pour $1 \leq i, j \leq n$ (voir : [Cho83]).

En particulier, $H^{(n)-1} \in M_n(\mathbb{Z})$. Ainsi, les matrices $H^{(n)}$ fournissent un exemple de matrices dont on peut calculer explicitement et formellement les coefficients de l'inverse, mais dont le calcul numérique (approchée) de l'inverse est difficile du fait de leur grand conditionnement.

Afin d'illustrer ce résultat, nous avons demandé à Sage [S⁺11] de calculer l'inverse de $H^{(20)}$ en prenant ses coefficients comme des nombres réels flottants avec double précision. Pour le coefficient d'indice (1, 1) de l'inverse, la réponse pour le calcul numérique est $x_{\mathbb{R},num} = 148,7 + O(10^{(-2)})$, tandis que la réponse exacte est $x_{exact} = 400$. En d'autres termes, $|x_{\mathbb{R},num} - x_{exact}|_{\mathbb{R}} > 250$.

Comportement p -adique

Nous allons maintenant voir que les calculs numériques en p -adique sur les matrices de Hilbert sont bien plus aisés qu'en réel, et le problème du conditionnement n'a plus lieu. En effet, nous pouvons écrire le lemme suivant :

Lemme 1.3.17. *Nous avons l'encadrement suivant sur les valuations des facteurs invariants de $H^{(n)}$:*

$$-\log_p(2n) = \sigma_1(H^{(n)}) \leq \sigma_n(H^{(n)}) \leq 0.$$

Démonstration. La minoration sur σ_1 est claire. La majoration sur σ_n vient du fait que $H^{(n)-1} \in M_n(\mathbb{Z})$. \square

Vue la Proposition 1.3.11, nous pouvons en déduire que le calcul de l'inverse de $H^{(n)}$ en passant par \mathbb{Q}_p produit un gain de précision de $-\sigma_n(H^{(n)})$. Ainsi, alors que le calcul de l'inverse de $H^{(n)}$ par des méthodes numériques réelles est instable du fait du conditionnement de $H^{(n)}$, le calcul par des méthodes numériques p -adiques produit même un gain de précision (absolue).

Par exemple, pour le calcul de l'inverse de $H^{(20)}$ en prenant ses coefficients dans \mathbb{Q}_2 avec précision absolue $O(2^{20})$, nous obtenons pour le coefficient d'indice (1, 1) de $H^{(20)-1}$ $x_{\mathbb{Q}_2,num} = 400 + O(2^{24})$. Ainsi, $|x_{\mathbb{Q}_2,num} - x_{exact}|_{\mathbb{Q}_2} \leq \frac{1}{2^{24}}$.

Si l'on applique un algorithme de reconstruction rationnelle (voir [VZGG13] Section 5.10) à $x_{\mathbb{Q}_2,num}$, on retrouve évidemment le coefficient $x_{exact} = 400$. Pour $p = 2$ et $H^{(20)}$, nous avons constaté qu'il suffisait à Sage d'une précision 130 pour que la reconstruction rationnelle retrouve tous les coefficients de $H^{(20)-1}$. Pour $p = 11$, une précision 60 suffit.

Perspectives

Une conséquence de l'étude précédente est qu'il paraît tout à fait raisonnable, lorsqu'on s'intéresse à des opérations comme l'inversion de matrices à coefficients rationnels, d'effectuer les calculs dans \mathbb{Q}_p puis d'appliquer un algorithme de reconstruction rationnelle. De manière informelle, pour une matrice M donnée, peu de p impliquent un $\sigma_n(M)$ grand. Génériquement (en un certain sens), même pour des matrices dont le conditionnement réel est grand, l'inversion se fait de manière stable.

Une telle approche est développée, par exemple dans [LL14]. Les auteurs de cet article proposent de plus d'effectuer les calculs en parallèle pour plusieurs p différents, puis de combiner théorème des restes chinois et reconstruction rationnelle. Ainsi, même les p induisant une valuation $\sigma_n(L)$ grande ne posent plus de problème, et de tels calculs avec plusieurs p ont l'avantage d'être totalement indépendants, donc très efficacement et facilement parallélisables.

1.4. Les limites des méthodes directes

Malgré les résultats positifs que peuvent apporter le suivi pas à pas avec les Propositions 1.1.1 et 1.1.2, cette méthode n'est pas optimale, et produit souvent des résultats très décevants par rapport

à ce que l'on pourrait théoriquement être en droit d'espérer. Nous présentons dans cette Section quelques exemples illustrant les échecs de cette méthode de suivi direct pas à pas. Ceci constituera une motivation pour l'introduction de méthodes plus puissantes de suivi de la précision.

1.4.1. Un exemple naïf

Considérons la fonction $f : \mathbb{Q}_p^2 \rightarrow \mathbb{Q}_p^2$ envoyant (x, y) sur $(x + y, x - y)$ et le problème du calcul de $f \circ f$ en $(a + O(p^n), b + O(p^m))$. En appliquant f consécutivement deux fois, et suivant la précision à chaque étape, nous obtenons $(2a + O(p^{\min(m,n)}), 2b + O(p^{\min(m,n)}))$. D'un autre côté, $f \circ f(x, y) = (2x, 2y)$, donc il est possible de calculer le résultat plus précisément comme $(2a + O(p^n), 2b + O(p^m))$ (et même une meilleure précision si $p = 2$).

Cette exemple est le symbole de deux travers qui gênent le suivi pas à pas de la précision :

- le premier est que le suivi pas à pas ne sait pas ce qu'il calcule : beaucoup d'information est perdue lorsqu'on applique quatre fois la formule sur la précision d'une somme sans savoir que ce que l'on calcule est $f \circ f$.
- le suivi pas à pas n'est pas bon pour gérer des ordres de grandeurs différents, en particulier en dimension supérieur ou égal à 2. C'est en particulier sur cet exemple où si l'on prenait $n = 1$ et $m = 10$, le suivi pas à pas nous donnerait une précision $O(p^1)$ sur les deux composantes, contre une précision $O(p^{10})$ qui peut être attendue sur la seconde.

1.4.2. D'autres exemples

Le déterminant

Le calcul du déterminant fournit un bon exemple pour montrer qu'un suivi pas à pas naïf peut ne pas fournir de réponse satisfaisante. En effet, soit $M \in M_n(\mathbb{Z}_p)$ une matrice dont les coefficients sont tous connus avec précision $O(p^m)$: si nous appliquons le Théorème 1.2.6, nous pouvons montrer qu'il est possible de calculer un déterminant grâce à un échelonnement en ligne, et que la perte de précision dans ce calcul est donnée par la valuation de ce déterminant.

Si nous appliquons maintenant, le calcul d'une forme normale de Smith approchée par l'Algorithme 1.3.5, la Proposition 1.3.7 nous donne que nous pouvons calculer $\det M$ à précision au moins $O(p^m)$ (et certainement mieux, mais cela sera précisé en Sous-Section 3.2.2). Ce résultat pourrait aussi s'obtenir en considérant la formule explicite définissant le déterminant en fonction des coefficients : $\det M = \sum_{\sigma \in \mathfrak{S}_n} \varepsilon(\sigma) M_{i\sigma(i)}$. Notons par ailleurs qu'il existe des algorithmes polynomiaux et sans division pour calculer le déterminant (voir [Ber84]), ce qui aurait permis de retrouver directement le fait que dans ce cas, la précision sur le déterminant est au moins $O(p^m)$.

La suite SOMOS-4

Cet exemple accompagnera toute notre étude de la précision p -adique. Il constitue un exemple jouet non-trivial où une approche naïve n'est pas suffisante et où toutes nos constructions s'appliquent directement.³

Définition 1.4.1. La suite SOMOS-4 [Som89] est définie par la relation de récurrence :

$$u_n = \frac{u_{n-3}u_{n-1} + u_{n-2}^2}{u_{n-4}}.$$

Cette relation apparaît lorsqu'on applique une formule d'addition dans une certaine courbe elliptique. Nous nous intéresserons spécifiquement au cas où les coefficients initiaux u_0, u_1, u_2 et u_3 sont dans \mathbb{Z}_p^\times et connus à précision $O(p^N)$.

Regardons tout d'abord ce qu'il advient de la précision lorsqu'on applique la formule de récurrence pour calculer u_n à partir de u_{n-4}, \dots, u_{n-1} . Comme on effectue deux multiplications et une addition, puis une division par u_{n-4} , la Proposition 1.1.2 nous donne que si les u_{n-4}, \dots, u_{n-1} sont connus à précision $O(p^l)$, alors u_n est connu à précision $O(p^{l-\text{val}(u_{n-4})})$. Ainsi une telle analyse pas à pas

3. À notre connaissance, ce sont Buhler et Kedlaya, avec des exposés à diverses conférences, qui ont les premiers montrés l'intérêt de cet exemple pour le suivi de la précision p -adique.

1. Méthode directe, applications et limites

nous donne pour le calcul de u_n par applications successives de la formule de récurrence en partant de u_0, u_1, u_2 et u_3 dans \mathbb{Z}_p^\times et connus à précision $O(p^N)$:

$$O(p^{N-v_n}) \quad \text{avec} \quad v_n = \text{val}(u_0) + \dots + \text{val}(u_{n-4}). \quad (1.1)$$

D'un autre côté, il est possible de montrer, et ceci demande des outils combinatoires avancés, que la suite SOMOS-4 satisfait ce que l'on nomme le *phénomène de Laurent* [FZ02] : pour tout entier $n \in \mathbb{N}$, il existe un polynôme P_n dans $\mathbb{Z}[X^{\pm 1}, Y^{\pm 1}, Z^{\pm 1}, T^{\pm 1}]$ (et pas seulement dans $\mathbb{Q}(X, Y, Z, T)$!) tel que $u_n = P_n(u_0, u_1, u_2, u_3)$.

De cette formule et de notre hypothèse de départ que u_0, u_1, u_2 et u_3 sont inversibles dans \mathbb{Z}_p connus à précision $O(p^N)$, il vient directement que tous les u_n sont connus au moins à la même précision, $O(p^N)$. Ainsi, le v_n que nous avons défini dans (1.1) ne reflète pas le comportement intrinsèque de la précision sur cet exemple mais plutôt une instabilité numérique qui apparaît lorsqu'on suit la précision pas à pas.

Remarque 1.4.2. À partir de la discussion précédente, il est possible de déduire un algorithme stable de calcul des termes de la suite SOMOS-4 :

1. calculer les polynômes P_n en utilisant la formule de récurrence, dans l'anneau $\mathbb{Z}[X^{\pm 1}, Y^{\pm 1}, Z^{\pm 1}, T^{\pm 1}]$
2. évaluer P_n au point (u_0, u_1, u_2, u_3) .

Cependant, le calcul des P_n est bien sûr très coûteux en temps, puisqu'il demande une division dans un anneau de polynômes en 4 variables, et la taille des coefficients de P_n peut exploser lorsque n croît.

Dans la Sous-Section 4.3.1, nous produirons un algorithme pour calculer les termes de la suite SOMOS-4 qui est à la fois efficace concernant la complexité et numériquement stable.

Remarque 1.4.3. Bien d'autres exemples pourraient être donnés. Citons par exemple le cas de la décomposition LU des matrices, où dans [Car12], l'auteur montre que selon le choix d'algorithme (échelonnement sans choix du pivot ou algorithme plus subtil et stabilisé) les pertes de précision diffèrent.

Le nombre d'arrangements

Nous présentons ici un exemple jouet où la perte de précision donnée par les méthodes directes ne correspond pas à celle que l'on devrait trouver. Il s'agit de calcul d'une liste de nombres d'arrangements.

Définition 1.4.4. Soit $n \in \mathbb{N}$ et $k \in \mathbb{N}$. Nous définissons le nombre d'arrangements A_n^k par $\frac{n!}{(n-k)!}$ si $n \geq k$ et 0 sinon. Il correspond au nombre d'applications injectives d'un ensemble de cardinal k vers un ensemble de cardinal n .

De par la définition, nous pouvons déduire la formule de récurrence suivante sur les nombres d'arrangements :

Lemme 1.4.5. Soit $n \in \mathbb{N}^*$ et $k \in \mathbb{N}^*$ avec $n \geq k$. Nous avons

$$A_{n+1}^k = \frac{n+1}{n-k+1} A_n^k.$$

Le problème qui nous intéresse est le suivant : pour un $k \in \mathbb{N}$ et un $m \in \mathbb{N}$ donnés, nous souhaitons calculer tous les termes de la liste $[A_0^k, \dots, A_m^k]$. Nous ajoutons une contrainte de plus : les entiers plus grands que k ne nous sont donnés qu'à précision $O(p^N)$ pour un certain p premier et $N \in \mathbb{N}^*$. Avec cette condition, nous voulons donc calculer les $[A_0^k, \dots, A_m^k]$ à la meilleure précision (p -adique) possible.

Remarque 1.4.6. Cette question s'applique naturellement au contexte suivant : on souhaite calculer la dérivée k -ème d'un polynôme de $\mathbb{Z}_p[X]$ degré m dont les coefficients sont connus à précision $O(p^N)$, et tel que m est très grand devant k , lui même grand devant 1.

À partir de la formule $A_n^k = \frac{n!}{(n-k)!}$, il est clair qu'il est possible de connaître A_n^k dans ce contexte à une précision au moins $O(p^N)$. Cependant, appliquer cette formule pour chacun des A_n^k à calculer veut dire k multiplications d'entiers de taille $\log(n)$, ce qui amène un temps de calcul que l'on souhaite éviter.

En conséquence, nous allons plutôt chercher à appliquer la formule de récurrence du Lemme 1.4.5. Mais si nous appliquons directement les Propositions 1.1.1 et 1.1.2, nous obtenons le résultat suivant :

Lemme 1.4.7. *Si $a, b, e \in \mathbb{Z}_p$ sont tels que $\text{val}(a) > \text{val}(b)$, et si $c \in \mathbb{N}^*$, alors*

$$(e + O(p^c) \times \frac{a + O(p^c)}{b + O(p^c)}) = aeb^{-1} + O(p^{c+\min(\text{val}(e)-\text{val}(b), \text{val}(a)-\text{val}(b))}).$$

En particulier, dès que $\text{val}(e) < \text{val}(b)$, de la précision absolue est perdue sur l'opération précédente.

Ce lemme a la conséquence suivante, dès que $\text{val}(n+1) < \text{val}(n-k+1)$, de la précision est perdue lors du calcul de A_{n+1}^k à partir de A_n^k . C'est un phénomène que l'on peut constater en pratique : si l'on implémente le calcul par récurrence avec des $i + O(2^N)$ en Sage [S⁺11], nous trouvons que pour $N = 12$, A_{19}^3 apparait connu seulement à précision $O(2^9)$.

Nous verrons par la suite qu'il est possible d'adapter le calcul par récurrence des A_n^k pour éviter ces phénomènes de perte de précision.

2. Le lemme de précision

"WELCOME TO WARP ZONE!"

Super Mario Bros

"It's super effective!"

Pokémon

Ce chapitre présente un travail commun avec Xavier Caruso et David Roe (voir [CRV14]). Dans celui-ci, nous présentons le contexte et les outils qui fondent la précision différentielle. Les premiers d'entre eux sont les réseaux, et l'objet de la Section 2.1 est de présenter et de motiver ces objets comme modèle de précision. La Section 2.2 présente elle notre lemme principal : il montre que pour des réseaux assez petits, travailler au premier ordre est suffisant pour pouvoir suivre la précision. La Section 2.3 fournit des outils de nature analytique pour estimer précisément quand le premier ordre est suffisant pour suivre la précision. La Section 2.4 présente plusieurs idées pour mettre en place en pratique le suivi de précision présenté dans les Sections précédentes. Enfin, la Section 2.5 montre que le cadre précédent, d'abord exprimé dans des espaces affines, est aussi disponible sur des variétés.

2.1. Les réseaux comme modèle de précision

2.1.1. Réseaux

L'une des idées principales défendues dans cette partie sur la précision p -adique est que la notion de réseau est la bonne pour ce qui est de la gestion de la précision. Il y a plusieurs raisons à cette opinion. La première est, déjà, que toutes les boules (centrées en zéro) dans un espace vectoriel ultramétrique sont des réseaux, et ainsi les réseaux ne font que généraliser la notion de boule. Cette généralisation a, cependant, un avantage décisif : l'image d'une boule par une application linéaire (invertible) n'est pas nécessairement une boule, tandis que l'image d'un réseau est bien un réseau. Ainsi, si l'on ne prend que des boules (centrées en zéro) comme donnée de précision, et même éventuellement une par composante, beaucoup d'information est perdue lorsque pour $A \in M_n(\mathbb{Z}_p)$ et $x \in \mathbb{Z}_p^n$, on veut calculer $A \cdot (x_1 + O(p^{N_1}), \dots, x_n + O(p^{N_n}))$ et ainsi, on doit donner une réponse sous la forme $(y_1 + O(p^{M_1}), \dots, y_n + O(p^{M_n}))$ (voir l'exemple en Sous-Section 1.4.1). Enfin, une troisième raison à l'utilisation de réseaux est qu'elle est particulièrement bien adaptée au travail en dimension infinie, en particulier sur des espaces de séries comme $\mathbb{Q}_p[[T]]$.

Avec ces motivations, rappelons que nous avons défini dans nos notations un *réseau* dans un K -espace de Banach E comme un sous- \mathcal{O}_K -module de E ouvert borné. Nous rappelons que tout réseau H de E est aussi fermé puisque son complémentaire est l'union des ensembles $a + H$ (avec $a \notin H$) qui sont tous ouverts.

Afin de se donner une première idée de ce à quoi ressemblent les réseaux dans un K -espace de Banach, nous en présentons une famille particulière qui est remarquable : celle des **réseaux diagonaux**. Ils sont définis lorsque E possède une base de Banach. Soit I un ensemble. Une famille libre $(x_i)_{i \in I} \subset E$ est une *base de Banach* pour E si tout élément $x \in E$ peut être écrit $x = \sum_{i \in I} \alpha_i x_i$ pour des scalaires $\alpha_i \in K$ avec $\alpha_i \rightarrow 0$ (suivant le filtre complémentaire des parties finies), et $\|x\| = \sup_{i \in I} |\alpha_i|$. Notons que si I est fini, la condition $\alpha_i \rightarrow 0$ est vide.

2. Le lemme de précision

Étant donnée une base de E $(x_i)_{i \in I}$ et une suite $(r_i)_{i \in I}$ avec $r_i \in \mathbb{R}_{>0}$, les ensembles

$$B_E((x_i), (r_i)) = \left\{ \sum_{i \in I} \alpha_i x_i : |\alpha_i| \leq r_i \right\},$$

$$B_E^-(((x_i), (r_i))) = \left\{ \sum_{i \in I} \alpha_i x_i : |\alpha_i| < r_i \right\}$$

sont des réseaux précisément lorsque les r_i sont bornés. Nous les appelons **réseaux diagonaux** de E . Si nous avons choisi une base privilégiée sur E , nous pouvons enlever (x_i) de la notation $B_E^{(-)}((x_i), (r_i))$.

Éléments approchés

Supposons que E est un K -espace de Banach ayant pour base $(x_i)_{i \in I}$.

Définition 2.1.1. — Un élément $x \in E$ est dit *exact* s'il existe un sous-ensemble $J \subseteq I$ et des scalaires $\alpha_j \in R$ tels que

$$x = \sum_{j \in J} \alpha_j x_j. \quad (2.1)$$

— Un *élément approché* est un couple (x, H) où $x \in E$ est un élément exact et H est un réseau de E .

Le couple (x, H) représente un élément indéterminé de l'ensemble $x + H$. Nous noterons souvent $x + O(H)$ pour insister sur le fait que H représente l'incertitude sur la valeur de l'élément approché. Dans le cas particulier de $E = K = \mathbb{Q}_p$, nous retrouvons la notion classique déjà introduite de $a + O(p^n)$ pour un élément approché p -adique. Remarquons que l'ensemble des éléments exacts est dense dans E , ainsi, tout élément de E peut être approché.

Réseaux et calcul effectif

Supposons que $E \simeq K^d$ est de dimension finie. Alors si $H \subset E$ est un réseau, il existe $a, b \in \mathbb{Q}_{>0}$ avec

$$B_K(a)^d \subset H \subset B_K(b)^d. \quad (2.2)$$

Notons $r = \frac{a}{b}$ et $R_r = \mathcal{O}_K / B_K(r)$. Alors un réseau H satisfaisant (2.2) est uniquement déterminé par son image sur le quotient $B_K(b)^d / B_K(a)^d \simeq R_r^d$. Comme $R \cap \mathcal{O}_K$ est dense dans \mathcal{O}_K , les éléments de R_r peuvent être représentés exactement. Ainsi, H peut être représenté comme une matrice $(d \times d)$ avec coefficients dans R_r . Par exemple, lorsque $K = \mathbb{Q}_p$, l'anneau R_r est simplement $(\mathbb{Z}/p^n\mathbb{Z})$ pour $n = \lfloor -\log_p r \rfloor$.

2.1.2. Séparer précision et approximation

La Définition 2.1.1 contient les deux principales thèses de ce chapitre concernant la représentation des espaces vectoriels, matrices, polynômes et séries sur K :

1. **Séparer** approximation et précision,
2. Les objets appropriés pour représenter la précision sont les **réseaux**.

Ces choix ont de nombreux avantages.

Remarquons tout d'abord qu'utiliser des réseaux pour représenter la précision sur des éléments approchés peut réduire la perte de précision en comparaison de stocker séparément la précision sur chaque coefficient α_i de (2.1). Reprenons l'exemple de la Sous-Section 1.4.1 $f : (x, y) \mapsto (x+y, x-y)$, et écrivons (e_1, e_2) pour la base canonique de $E = \mathbb{Q}_p^2$. Comme f est linéaire, l'image de l'approximation $((a, b), B_E((e_1, e_2), (p^{-n}, p^{-m})))$ est $((a+b, a-b), B_E((e_1+e_2, e_1-e_2), (p^{-n}, p^{-m})))$. Pour $p \neq 2$, en appliquant f à nouveau, nous obtenons $((2a, 2b), B_E((e_1, e_2), (p^{-n}, p^{-m})))$. Ainsi, utiliser des réseaux élimine complètement la perte de précision vue en Sous-Section 1.4.1. Nous verrons dans la Section suivante qu'un phénomène similaire se produit aussi pour des applications qui ne sont pas linéaires.

En plus de permettre des représentations plus souples de la précision sur un élément, séparer la précision de l'approximation a aussi d'autres avantages. Si la précision est encodée avec une approximation, certains algorithmes deviennent inutilisables du fait de leur instabilité numérique. Par exemple, l'algorithme de Karatsuba pour la multiplication de polynômes [KO62] peut perdre plus de précision qu'il ne devrait lorsqu'il est utilisé sur des polynômes à coefficients inexacts. Cependant, il fonctionne parfaitement sur des approximations qui sont des éléments exacts, laissant la question de la précision sur le produit être traitée séparément. Ainsi, en séparant la précision de l'approximation, plus d'algorithmes peuvent être utilisés.

2.1.3. Types de précision : quelques réseaux particuliers

Dans ce chapitre, nous allons voir qu'utiliser des réseaux pour suivre la précision lors de calculs sur des éléments approchés permet de diminuer les pertes de précision lors de ces calculs et ainsi, de permettre pour une même précision en sortie, une précision plus faible au cours des calculs intermédiaires, ce qui diminue le coût des calculs. Cependant, travailler avec des réseaux de manière exacte, bien que possible, peut coûter plus cher en espace. Par exemple, l'espace nécessaire pour stocker un réseau de précision pour une matrice $n \times n$ dont les coefficients sont connus à précision $O(p^N)$ est $O(Nn^4 \cdot \log p)$. D'un autre côté, l'espace nécessaire pour décrire que toutes les entrées sont connues à précision $O(p^N)$ est de seulement $O(\log N)$.

Pour pallier à ce problème, on peut vouloir choisir de se restreindre à une famille de réseaux particulière, pour laquelle représentation et calculs seraient moins coûteux que dans le cas général.

Définition 2.1.2. Supposons que E est un K -espace de Banach, et écrivons $\text{Lat}(E)$ pour l'ensemble des réseaux de E . Nous définissons un *type de précision* pour un K -espace de Banach E comme un ensemble $\mathcal{T} \subseteq \text{Lat}(E)$ avec une application de projection $\text{proj}_{\mathcal{T}} : \text{Lat}(E) \rightarrow \mathcal{T}$ telle que :

- (*) Pour tout réseau $H \in \text{Lat}(E)$, le réseau $\text{proj}_{\mathcal{T}}(H)$ est un plus petit majorant, dans \mathcal{T} , de H pour l'ordre défini par l'inclusion : $H \subseteq \text{proj}_{\mathcal{T}}(H)$ et si $T \in \mathcal{T}$ est tel que $T \subsetneq \text{proj}_{\mathcal{T}}(H)$ alors $H \not\subseteq T$.

En fonction des problèmes rencontrés, différents types de précision peuvent naturellement apparaître. Par exemple, lors de la dernière étape de l'algorithme de Kedlaya pour le calcul de fonctions zeta de courbes hyperelliptiques (voir [Ked01]*§4 : Step 3), on calcule le polynôme caractéristique de la matrice définie par l'action du morphisme de Frobenius sur un espace de cohomologie p -adique. Obtenir des chiffres de précision supplémentaires sur les entrées de la matrice demande un long calcul, donc on souhaite travailler avec un type de précision qui ne nous en fait pas trop perdre lorsqu'on projette sur lui. On souhaite ainsi que le type de précision soit adapté aux réseaux que l'on manipule au cours de cet algorithme.

La liste suivante donne quelques exemples de types de précisions utiles.

- La précision avec les **réseaux**, utilisant tous les réseaux, correspond à $\mathcal{T} = \text{Lat}(E)$.
- Le type de précision **diagonal**¹, correspond aux réseaux de la forme $\bigoplus_i \pi^{k_i} e_i$ (pour une base de Banach donnée (e_i) dans E). Ceci correspond à traiter avec des $O(\pi^{k_i})$ sur chaque composante dans l'écriture en fonction des e_i .
- Pour le type de précision **plat**, \mathcal{T} est constitué des réseaux $B_E(r)$. Autrement dit, on ne considère que les réseaux de la forme $\bigoplus_i \pi^{k_i} e_i$ (pour une base de Banach donnée (e_i) dans E). Ce type de précision peut être utile car il est très peu coûteux en espace et permet des calculs faciles. Par contre, on peut perdre beaucoup en projetant sur lui : $\text{proj}_{\mathcal{T}}(H)$ est la plus grande boule centrée en zéro contenant H .
- Si $E = K_{<d}[X]$ est l'espace des polynômes de degré strictement moins que d , le type de précision de **Newton** est constitué des réseaux de la forme $B_E((X^i), (r_i))$ avec $-\log r_i$ une fonction convexe de i . Ceci correspond à travailler seulement avec des polygones de Newton. Ceci est adapté si l'on pense aux polynômes comme des fonctions : une précision supplémentaire sur les coefficients des monômes strictement au-dessus du polygone de Newton ne modifie pas la précision connue sur l'évaluation.
- Si $E = M_{m \times n}(K)$, le type de précision **colonne** est constitué des réseaux qui, individuellement, ont la même image selon les projections $\text{pr}_i : E \rightarrow K^m$ envoyant une matrice sur sa i -ème

1. Nous avons utilisé le terme *jagged* dans [CRV14]

2. Le lemme de précision

colonne. Il est alors approprié lorsqu'on s'intéresse à des applications linéaires l'image de chaque vecteur d'une base est connu à la même précision selon les réseaux.

- Si $E = \mathbb{Q}_p[[X]]$, le type de précision de **Pollack-Stevens** est constitué des réseaux de la forme $H_N := B_E((X^i), (p^{\min(i-N, 0)}))$ [PS11]*§1.5. Ces réseaux sont stables sous l'action de certains opérateurs de Hecke, ce qui est nécessaire pour calculer des symboles modulaires surconvergents.

Remarquons que dans certains cas, il est possible de connaître le réseau de précision sur le résultat final *a priori*, en utilisant les méthodes de ce chapitre. Ce sera le cas sur les exemples des Chapitres 3 et 5. Connaître ce résultat permettra de minimiser les pertes de précision indues, et ce même lorsqu'on travaille avec des types de précision assez grossiers comme les types plats ou diagonaux. Séparer la précision de l'approximation rend aussi plus facile l'implémentation d'algorithmes capables de manipuler plusieurs types de précision différents. Il suffit dans ce cas de gérer l'arithmétique sur les approximations séparément de celle sur la précision.

2.1.4. Diffusion de la précision

Dans ce qui suit, en particulier au Chapitre 3, nous comparons l'usage de réseaux pour suivre la précision par rapport à la méthode classique de suivi direct coordonnée par coordonnée. La définition suivante sera alors utile pour mesurer la différence entre ces deux méthodes. Supposons que E est muni d'une base (e_1, \dots, e_n) et écrivons $\pi_i : E \rightarrow Ke_i$ pour les projections. Remarquons tout d'abord que suivre la précision coordonnée par coordonnée ne permet de considérer que des données de précision de la forme $\sum_{i=1}^n B_i e_i$ où les B_i sont des réseaux en dimension 1, *i.e.* des boules. En conséquence, ces données correspondent à des réseaux que nous pouvons qualifier de *diagonaux*. Définissons alors le nombre de chiffres de précision diffusés :

Définition 2.1.3. Si $H \subset E$ est un réseau, soit

$$H_0 = \pi_1(H) \oplus \dots \oplus \pi_n(H).$$

Soit $k = \log_p([H_0 : H])$. Alors nous disons que H a k chiffres de précision diffusés.

Si H représente la précision effective sur un objet, alors H_0 est la meilleure approximation dans le formalisme coordonnée par coordonnée de H . Avec le formalisme de la section précédente, H_0 est le projeté sur le type de précision diagonal. Il s'agit en effet du plus petit réseau diagonal contenant H . k représente alors bien le nombre de chiffres de précision perdus lorsqu'on se restreint à ne considérer que des réseaux diagonaux, ce que fait le cadre classique du suivi direct coordonnée par coordonnée.

Grâce aux méthodes présentées dans ce chapitre, nous allons voir au Chapitre 3 que le nombre de chiffres de précision diffusés peut parfois être loin d'être négligeable.

2.2. Lemme principal : réseaux du premier ordre et précision

2.2.1. Applications différentiables

Notre but dans cette Section est de relier l'image d'un réseau par une application différentiable à celle de son image par sa dérivée. Nous verrons que pour des réseaux assez petits, les deux sont directement liées. Pour cela, nous choisissons la définition suivante de différentiabilité, issue de l'analyse ultramétrique. Nous renvoyons à Schneider [Sch11] pour une introduction plus complète à ce domaine.

Définition 2.2.1. Soit E et F deux K -espaces de Banach, soit U un ouvert de E et soit $f : U \rightarrow F$ une application. Alors f est dit *différentiable* en $v_0 \in U$ s'il existe une application linéaire continue $f'(v_0) : U \rightarrow F$ telle que pour tout $\varepsilon > 0$, il existe un voisinage ouvert $U_\varepsilon \subset U$ contenant v_0 tel que

$$\|f(v) - f(w) - f'(v_0)(v - w)\| \leq \varepsilon \|v - w\|.$$

pour tout $v, w \in U_\varepsilon$. L'application linéaire $f'(v_0)$ est appelée la *différentielle* de f en v_0 .

Remarque 2.2.2. Cette notion de différentiabilité est parfois appelée *stricte différentiabilité*. Elle implique que la fonction $x \mapsto f'(x)$ est continue sur U .

Cette notion de différentiabilité est mieux adaptée à l'analyse dans un contexte ultramétrique que la définition classique, réelle ou complexe, à partir de taux d'accroissements. En effet, en prenant cette dernière, on trouve des exemples de fonctions de \mathbb{Q}_p dans \mathbb{Q}_p qui sont injectives, différentiables et de différentielle nulle ! Ce type de problème n'apparaît pas avec la définition que nous avons choisie. Nous renvoyons à [Rob00] pour plus de détails sur les différentes notions de différentiation dans un contexte ultramétrique.

La Définition 2.2.1 implique toutes les propriétés usuelles que l'on attend d'une dérivation : unicité, composition, correspondance avec la dérivée formelle sur les polynômes ou les séries convergentes, etc (voir [Sch11] et [Rob00]).

2.2.2. Images de réseaux par des applications différentiables

Définition 2.2.3. Soit E et F deux K -espaces de Banach, $f : U \rightarrow F$ une fonction définie sur un ouvert U de E et v_0 un point de U . Un réseau H de E est appelé un *réseau du premier ordre* pour f en v_0 si $v_0 + H \subset U$ et que l'on a l'égalité suivante :

$$f(v_0 + H) = f(v_0) + f'(v_0)(H). \quad (2.3)$$

Nous insistons sur le fait que nous demandons à l'égalité (2.3) d'être une égalité, et pas seulement une inclusion ! Avec cette définition, nous pouvons énoncer notre lemme principal :

Lemme 2.2.4. Soit E et F deux K -espaces de Banach et $f : U \rightarrow F$ une application définie sur un ouvert U de E . Nous supposons que f est différentiable en un point $v_0 \in U$ et que la différentielle $f'(v_0)$ est surjective.

Alors, pour tout $\rho \in]0, 1]$, il existe un réel positif δ tel que, pour tout $r \in]0, \delta[$, tout réseau H tel que $B_E^-(\rho r) \subset H \subset B_E(r)$ est un réseau du premier ordre pour f en v_0 .

Démonstration. Sans perdre en généralité, nous pouvons supposer que $v_0 = 0$ et $f(0) = 0$. Comme $f'(0)$ est surjective, le théorème de l'application ouverte fournit un $C > 0$ tel que $B_F(1) \subset f'(0)(B_E(C))$. Soit $\varepsilon > 0$ tel que $\varepsilon C < \rho$, et soit $U_\varepsilon \subset E$ donné comme dans la Définition 2.2.1 par le fait que f soit différentiable en v_0 . Nous pouvons supposer que $U_\varepsilon = B_E(\delta)$ pour un certain $\delta > 0$.

Soit $r \in]0, \delta[$. Soit H un réseau tel que $B_E^-(\rho r) \subset H \subset B_E(r)$. Nous voulons montrer que f envoie H surjectivement sur $f'(0)(H)$. Nous montrons d'abord que $f(H) \subset f'(0)(H)$. Supposons que $x \in H$. En vertu de la différentiabilité de f en 0, $\|f(x) - f'(0)(x)\| \leq \varepsilon \|x\|$. Posons $y = f(x) - f'(0)(x)$, nous avons $\|y\| \leq \varepsilon r$. La définition de C implique que $B_F(\varepsilon r) \subset f'(0)(B_E(\varepsilon r C))$. Ainsi, il existe $x' \in B_E(\varepsilon r C)$ tel que $f'(0)(x') = y$. Comme $\varepsilon C < \rho$, nous obtenons $x' \in B_E^-(\rho r) \subset H$ et alors $f(x) = f'(0)(x - x') \in f'(0)(H)$.

Prouvons maintenant la surjectivité. Soit $y \in f'(0)(H)$. Soit $x_0 \in H$ tel que $y = f'(0)(x_0)$. Nous construisons par récurrence deux suites, (x_n) et (z_n) telles que :

- $z_n \in E$ satisfait $f'(0)(z_n) = y - f(x_n)$ et $\|z_n\| \leq C \cdot \|y - f(x_n)\|$ (un tel élément existe par définition de C), et
- $x_{n+1} = x_n + z_n$.

Nous posons de plus $x_{-1} = 0$ et $z_{-1} = x_0$. Nous montrons que les suites (x_n) et (z_n) sont bien définies et prennent leurs valeurs dans H . Nous pouvons le faire par récurrence : ce fait est vrai pour $x_{-1} = 0$ et $z_{-1} = x_0$, et si x_{n-1} et x_n appartiennent à H , nous montrons qu'il en est de même de z_n et x_{n+1} . Remarquons en effet que :

$$\begin{aligned} y - f(x_n) &= f(x_{n-1}) + f'(0)(z_{n-1}) - f(x_n) \\ &= f(x_{n-1}) - f(x_n) - f'(0)(x_{n-1} - x_n). \end{aligned} \quad (2.4)$$

En utilisant la différentiabilité, nous en déduisons que $\|y - f(x_n)\| \leq \varepsilon \cdot \|x_n - x_{n-1}\|$. Comme nous avons supposé que x_{n-1} et x_n sont dans $H \subset B_E(r)$, nous obtenons $\|y - f(x_n)\| \leq \varepsilon r$. Ainsi $\|z_n\| \leq C \cdot \varepsilon r < \rho r$ et donc $z_n \in H$. Puisque $x_{n+1} = x_n + z_n$, nous en déduisons finalement que $x_{n+1} \in H$.

2. Le lemme de précision

En utilisant (2.4) et la différentiabilité en 0 une fois de plus, nous obtenons

$$\|y - f(x_n)\| \leq \varepsilon \cdot \|z_{n-1}\| \leq \varepsilon C \cdot \|y - f(x_{n-1})\|,$$

pour tout $n > 0$. En conséquence, $\|y - f(x_n)\| = O(a^n)$ et $\|z_n\| = O(a^n)$ pour $a = \varepsilon C < \rho \leq 1$. Ceci montre que (x_n) est une suite de Cauchy, et donc qu'elle converge puisque E est complet. Notons x la limite de (x_n) . Nous avons $x \in H$ puisque H est fermé. De plus, f est continue sur $H \subseteq U_\varepsilon$ puisque différentiable, et ainsi $y = f(x)$. Ceci conclut la démonstration. \square

Nous concluons cette Sous-Section avec une remarque concernant la surjectivité de $f'(v_0)$ que nous avons pris en hypothèse du Lemme 2.2.4. Tout d'abord, nous devons insister sur le fait que cette hypothèse est bien nécessaire. En effet, le lemme montrerait dans le cas contraire que l'image de f est localement contenue dans un sous-espace vectoriel strict au voisinage de chaque point où la différentielle de f n'est pas surjective, ce qui n'est bien sûr pas vrai. Il suffit de regarder $f : \mathbb{Q}_p \rightarrow \mathbb{Q}_p$, définie par $x \mapsto x^2$, en 0.

Néanmoins, nous pouvons utiliser le Lemme 2.2.4 pour prouver un résultat plus faible dans le contexte où $f'(v_0)$ n'est pas surjective. Pour cela, choisissons un sous-espace vectoriel fermé W de F tel que $W + f'(v_0)(E) = F$. Notons pr_W pour la projection canonique de F sur F/W . Alors la composée $\text{pr}_W \circ f$ est différentiable en v_0 avec une différentielle surjective. Pour un réseau donné H , il y a beaucoup de choix de W pour lesquels le Lemme 2.2.4 s'applique. Pour un tel W ,

$$f(v_0 + H) \subset f(v_0) + f'(v_0)(H) + W, \quad (2.5)$$

et on peut prendre l'intersection des membres de droite pour différents W , et ainsi, obtenir une borne supérieure sur $f(v_0 + H)$.

2.3. Caractérisation analytique des réseaux du premier ordre

2.3.1. Le cas des fonctions localement analytiques

Polygones de Newton et condition suffisante

Cette Section est consacrée à rendre le δ du Lemme 2.2.4 explicite, sous l'hypothèse supplémentaire que f est localement analytique. Nous étendons la définition de telles fonctions des K -espaces vectoriels de dimension finie dans [Sch11] §6 aux K -espaces de Banach.

Définition 2.3.1. Soit E et F deux K -espaces de Banach. Soit U un ouvert de E et soit $x \in U$. Une fonction $f : U \rightarrow F$ est dite *localement analytique* en x s'il existe un ouvert $U_x \subset E$ contenant x et des applications n -linéaires continues $L_n : E^n \rightarrow F$ pour tout $n \geq 1$ tels que

$$f(x + h) = f(x) + \sum_{n \geq 1} L_n(h, \dots, h)$$

pour tout h tel que $x + h \in U_x$.²

Remarque 2.3.2. Une fonction f qui est localement analytique en x est *a fortiori* différentiable en x , avec dérivée donnée par L_1 .

Dans le reste de cette Section, nous supposons que K est algébriquement clos. Comme dans la Définition 2.3.1, nous considérons deux K -espaces de Banach E et F et une famille d'applications n -linéaires continues $L_n : E^n \rightarrow F$. Pour $n \geq 1$ et $h \in E$, nous posons $f_n(h) = L_n(h, \dots, h)$ et

$$\|f_n\| = \sup_{h \in B_E(1)} \|f_n(h)\|.$$

Lorsque la série $\sum_n f_n(h)$ converge, nous notons par $f(h)$ sa somme. Nous noterons aussi $f = \sum_{n \geq 0} f_n$. Nous pouvons supposer que f est définie sur un voisinage de 0. Sous ces hypothèses, la

2. En particulier, nous demandons à la série dans le membre de droite d'être convergente.

donnée de f détermine uniquement les f_n (une conséquence de la Proposition 2.3.5 plus bas). À une telle série f , nous associons la fonction $\Lambda(f) : \mathbb{R} \cup \{+\infty\} \rightarrow \mathbb{R} \cup \{+\infty\}$ définie par :

$$\Lambda(f)(v) = \begin{cases} \log \left(\sup_{h \in B_E^-(e^v)} \|f(h)\| \right) & \text{si } f \text{ est défini sur } B_E^-(e^v), \\ +\infty & \text{sinon.} \end{cases}$$

Le lemme suivant s'obtient directement, et est laissé au lecteur.

Lemme 2.3.3. *Soit $f = \sum_{n \geq 0} f_n$ et $g = \sum_{n \geq 0} g_n$ deux séries comme précédemment. Alors*

$$\begin{aligned} \Lambda(f + g) &\leq \max(\Lambda(f), \Lambda(g)), \\ \Lambda(f \times g) &\leq \Lambda(f) + \Lambda(g), \\ \Lambda(f \circ g) &\leq \Lambda(f) \circ \Lambda(g). \end{aligned}$$

Remarque 2.3.4. En appliquant le Lemme 2.3.3, nous pouvons obtenir aisément une majoration sur $\Lambda(f)$ à partir d'une formule qui décrirait explicitement f .

La fonction $\Lambda(f)$ que nous venons de définir est liée de près au polygone de Newton de f . Rappelons que le polygone de Newton de f est l'enveloppe convexe inférieure dans \mathbb{R}^2 des points $(n, -\log\|f_n\|)$ pour $n \geq 0$, et du point $(0, +\infty)$. Nous notons $\text{NP}(f) : \mathbb{R} \rightarrow \mathbb{R} \cup \{+\infty\}$ la fonction convexe dont l'épigraph est le polygone de Newton de f .

Nous rappelons que la transformée de Legendre d'une fonction convexe $\varphi : \mathbb{R} \rightarrow \mathbb{R} \cup \{+\infty\}$ est la fonction $\varphi^* : \mathbb{R} \rightarrow \mathbb{R} \cup \{+\infty\}$ définie par

$$\varphi^*(v) = \sup_{u \in \mathbb{R}} (uv - \varphi(u)),$$

pour $v \in \mathbb{R}$. Nous remarquons que l'application $\varphi \mapsto \varphi^*$ est une involution renversant l'ordre : $(\varphi^*)^* = \varphi$ et $\varphi^* \geq \psi^*$ dès que $\varphi \leq \psi$. Si nécessaire, nous étendons φ^* à $\mathbb{R} \cup \{+\infty\}$ par continuité à gauche. Nous renvoyons à [Roc97] pour une introduction plus complète à la transformée de Legendre.

Proposition 2.3.5. *Gardant les notations précédentes, nous avons $\Lambda(f) = \text{NP}(f)^*$.*

Démonstration. Remarquons tout d'abord que les applications $\Lambda(f)$ et $\text{NP}(f)^*$ sont toutes deux continues à gauche. Il est alors suffisant de prouver qu'elles coïncident en-dehors de l'ensemble des pentes de $\text{NP}(f)$, i.e. sur un sous-ensemble dense de \mathbb{R} .

Soit v un nombre réel qui n'est pas une pente de $\text{NP}(f)$. Nous supposons d'abord que $\text{NP}(f)^*(v)$ est fini. Nous posons $u = \text{NP}(f)^*(v)$. La fonction $m \mapsto \text{NP}(f)(m) - vm + u$ a alors les propriétés suivantes :

1. elle est affine par morceaux et partout positive (ou nulle) ;
2. elle n'admet pas 0 comme pente ;
3. elle s'annule en $x = n$ pour un certain entier n et $u = vn + \log\|f_n\|$.

Nous pouvons déduire de ces faits qu'il existe un $c > 0$ tel que

$$vm - u \leq -\log\|f_m\| - c \cdot |n - m|$$

pour tout $m \geq 0$. Comme $vm - u = vm - vn - \log\|f_n\|$, nous obtenons

$$-vn - \log\|f_n\| + c \cdot |n - m| \leq -vm - \log\|f_m\|.$$

Alors, pour tout $x \in B_E(e^v)$ et $m \geq 0$, nous avons

$$\|f_m(x)\| \leq e^{-c \cdot |n-m|} \cdot \|f_n\| \cdot e^{vn} \leq \|f_n\| \cdot e^{vn}.$$

Ainsi, la série $\sum_{m \geq 0} f_m(x)$ converge et $\|f(x)\| \leq \|f_n\| \cdot e^{vn}$. Nous obtenons alors :

$$\Lambda(f)(v) \leq \log(\|f_n\| e^{vn}) = vn + \log\|f_n\| = u. \quad (2.6)$$

2. Le lemme de précision

Par ailleurs, il suit de la définition de $\|f_n\|$ et du fait que $|K^\times|$ est dense dans \mathbb{R} (K est algébriquement clos) qu'il existe une suite $(x_i)_{i \geq 0}$ de $B_E^-(e^v)$ telle que $\lim_{i \rightarrow \infty} \|f_n(x_i)\| = \|f_n\| \cdot e^{vn}$. Puisque $\|f_m(x_i)\| \leq e^{-c \cdot |n-m|} \cdot \|f_n\| \cdot e^{vn}$ pour tout m et i , nous obtenons $\|f_m(x_i)\| < \|f_n(x_i)\|$ pour i assez grand. Pour un tel i , nous avons alors $\|f(x_i)\| = \|f_n(x_i)\|$. En passant à la limite sur i , nous trouvons $\Lambda(f)(v) \geq u$. En comparant avec (2.6), nous obtenons $\Lambda(f)(v) = u = \text{NP}(f)^*(v)$.

Nous supposons maintenant que $\text{NP}(f)^*(v) = +\infty$. La fonction $x \mapsto \text{NP}(f)(x) - vx$ n'est alors pas minorée. Comme elle est convexe, elle tend vers $-\infty$ lorsque x tend vers $+\infty$. Vue la définition de $\text{NP}(f)$, l'expression $vn + \log \|f_n\|$ tend vers l'infini quand n tend vers l'infini. Il est alors suffisant de montrer le résultat suivant :

$$\forall n \in \mathbb{N}, \quad \Lambda(f)(v) \geq vn + \log \|f_n\| - \log 2. \quad (2.7)$$

Soit $n \in \mathbb{N}^*$. Si $\|f_n\| = 0$, il n'y a rien à montrer. Sinon, soit $x_n \in B_E^-(e^v)$ tel que $\|f_n(x_n)\| \geq \frac{1}{2} \|f_n\| \cdot e^{vn}$. Si la série $\sum_{m \geq 0} f_m(x_n)$ diverge, alors $\Lambda(f)(v) = +\infty$ par définition et l'équation (2.7) est satisfaite. D'un autre côté, si cette série converge, la suite $\|f_m(x_n)\|$ tend vers 0 quand m tend vers l'infini. Ainsi, elle atteint son maximum, M , un nombre fini de fois. Notons $I \subset \mathbb{N}$ l'ensemble des indices correspondant. Pour tout $\lambda \in \mathcal{O}_K$, la série définissant $f(\lambda x_n)$ converge et

$$f(\lambda x_n) \in B_F(M) \quad \text{and} \quad f(\lambda x_n) \equiv \sum_{m \in I} \lambda^m f_m(x_n) \pmod{B_F^-(M)}.$$

Le quotient $B_F(M)/B_F^-(M)$ est un espace vectoriel sur le corps résiduel k_K de K . Comme k_K est infini, il existe $\lambda \in \mathcal{O}_K$ tel que $\sum_{m \in I} \lambda^m f_m(x_n)$ ne s'annule pas sur $B_F(M)/B_F^-(M)$. Pour un tel λ , nous avons $\|f(\lambda x_n)\| = M \geq \frac{1}{2} \|f_n\| \cdot e^{vn}$. L'équation (2.7) est alors vérifiée. Ceci conclut la preuve de cette proposition. \square

Remarque 2.3.6. Il découle de la Proposition 2.3.5 que $\Lambda(f)$ est une fonction convexe.

Nous étudions maintenant l'effet de la troncation sur les séries : étant donné f comme précédemment et un entier $n_0 \in \mathbb{N}$, nous posons

$$f_{\geq n_0} = \sum_{n \geq n_0} f_n = f - (f_0 + f_1 + \cdots + f_{n_0-1}).$$

Par ailleurs, étant donné une fonction convexe $\varphi : \mathbb{R} \rightarrow \mathbb{R} \cup \{+\infty\}$ et un nombre réel v , nous définissons $\varphi_{\geq v} : \mathbb{R} \rightarrow \mathbb{R} \cup \{\pm\infty\}$ par

$$\varphi_{\geq v}(x) = \inf_{y \geq 0} (\varphi(x+y) - vy).$$

La fonction $\varphi_{\geq v}$ est le maximum en chaque point parmi les fonctions φ' avec $\varphi' \leq \varphi$ et $x \mapsto \varphi'(x) - vx$ croissante. Lorsque v est fixé, l'application $\varphi \mapsto \varphi_{\geq v}$ est croissante au sens suivant : si φ et ψ sont deux fonctions convexes telles que $\varphi \leq \psi$ alors $\varphi_{\geq v} \leq \psi_{\geq v}$.

Proposition 2.3.7. *Avec les notations précédentes, nous avons $\Lambda(f_{\geq n_0}) \leq \Lambda(f)_{\geq n_0}$ pour tout $n_0 \in \mathbb{N}$.*

Démonstration. Ceci est une conséquence directe de la Proposition 2.3.5 et du fait que les pentes de la transformée de Legendre d'une fonction convexe affine par morceaux f sont exactement les abscisses des points où f n'est pas différentiable. \square

Nous fournissons maintenant deux conditions suffisantes pour reconnaître de manière effective les réseaux du premier ordre.

Proposition 2.3.8. *Soit $f = \sum_{n \geq 0} f_n$ une fonction comme précédemment. Soit $C \in \mathbb{R}_{>0}$ tel que $B_F(1) \subset f_1(B_E(C))$. Soit $\rho \in]0, 1]$ et $\nu \in \mathbb{R}$ tel que*

$$\Lambda(f)_{\geq 2}(\nu) < \nu + \log \left(\frac{\rho}{C} \right). \quad (2.8)$$

Alors la conclusion du Lemme 2.2.4 est satisfaite pour $\delta = e^\nu$.

Remarque 2.3.9. Sur un voisinage de $-\infty$, la fonction $x \mapsto \Lambda(f)_{\geq 2}(x) - x$ est affine de pente 1. Ceci implique que, pour tout $\rho \in]0, 1]$, il existe ν satisfaisant (2.8). De plus, si ρ est assez proche de 0, alors on peut prendre $\delta = e^\nu$ comme une fonction linéaire de ρ .

Remarque 2.3.10. Dans l'énoncé de la Proposition 2.3.8, nous pouvons bien sûr remplacer la fonction $\Lambda(f)$ par n'importe quelle fonction convexe φ avec $\varphi \geq \Lambda(f)$. Si f est donnée par une formule ou un algorithme, une telle fonction φ peut être obtenue grâce à la Remarque 2.3.4.

Démonstration. Prenons ε dans l'intervalle ouvert $]e^{\Lambda(f)_{\geq 2}(\nu) - \nu}, \frac{\rho}{C}[$. En revenant à la preuve du Lemme 2.2.4, nous observons qu'il est suffisant de montrer que

$$\|f_{\geq 2}(x)\| \leq \varepsilon \cdot \|x\|. \quad (2.9)$$

pour tout $x \in B_E(\delta)$. Cette inégalité est une conséquence des Propositions 2.3.5 et 2.3.7 appliquées à la fonction $x \mapsto \frac{\Lambda_{\geq 2}(x)}{x}$. \square

Remarque 2.3.11. Il suit de la preuve que la Proposition 2.3.8 reste vraie si K n'est pas algébriquement clos. En effet, les fonctions f_n , et ainsi f , s'étendent à une clôture algébrique \bar{K} de K et (2.9) est satisfaite sur \bar{K} , ce qui suffit pour obtenir le résultat.

Corollaire 2.3.12. *Nous conservons les notations de la Proposition 2.3.8 et considérons de plus une suite $(M_n)_{n \geq 2}$ telle que $\|f_n\| \leq M_n$ pour tout $n \geq 2$. Soit $NP(M_n)$ la fonction convexe dont l'épigraphe est l'enveloppe convexe inférieure dans \mathbb{R}^2 des points de coordonnées $(n, -\log M_n)$ pour $n \geq 2$ et du point $(0, +\infty)$.*

Soit $\rho \in]0, 1]$ et $\nu \in \mathbb{R}$ tels que

$$NP(M_n)^*(\nu) < \nu + \log\left(\frac{\rho}{C}\right).$$

Alors la conclusion du Lemme 2.2.4 est satisfaite pour $\delta = e^\nu$.

Remarque 2.3.13. Si K est de caractéristique 0 et que les espaces vectoriels E et F sont de dimension finie, alors les M_n définis comme ceci :

$$M_n = \frac{1}{|n!|} \cdot \sup_{\substack{1 \leq i \leq \dim E \\ |\underline{n}|=n}} \left\| \frac{\partial^n f_i}{\partial x^{\underline{n}}}(0) \right\|$$

conviennent. Ici f_i signifie la i -ème coordonnée de f , la notation \underline{n} est prise pour les $(\dim F)$ -uplets d'entiers positifs et $|\underline{n}|$ est la somme des coordonnées de \underline{n} .

Un cas particulier

Nous énonçons ici une interprétation des résultats précédents dans le cas des fonctions polynomiales entières. Une fonction $f : E \rightarrow F$ est dite *polynomiale entière* si elle est donnée par un polynôme multivarié à coefficients dans \mathcal{O}_K dans un (ou de façon équivalente dans tout) système de coordonnées associé à une \mathcal{O}_K -base de $B_E(1)$.

Proposition 2.3.14. *Avec les notations du Lemme 2.2.4 et en supposant de plus que f est polynomial entier, soit $C \in \mathbb{R}_{>0}$ tel que $B_F(1) \subset f'(v_0)(B_E(C))$. Alors la conclusion du Lemme 2.2.4 est satisfaite avec $\delta = C \cdot \rho^{-1}$.*

Démonstration. Nous appliquons le Corollaire 2.3.12. Pour cela, nous pouvons prendre $M_n = 1$ du fait de la supposition que les coefficients de f sont dans \mathcal{O}_K . Ceci nous donne $NP(M_n)^*$ qui est la demi-droite d'équation $y = 0$ pour $x \geq 2$. Sa transformée de Legendre est alors $NP(M_n)^*$ d'équation $y = 2x$ pour $x \leq 0$ et $+\infty$ pour $x \geq 0$. On a alors directement que $NP(M_n)^*(\nu) = 2\nu < \nu + \log\left(\frac{\rho}{C}\right)$ dès que $\nu < \log\left(\frac{\rho}{C}\right)$ et $\nu \leq 0$. D'où le résultat. \square

2.3.2. Caractérisation par une équation différentielle

Dans les chapitres suivants, nous allons calculer la différentielle de certaines opérations classiques (déterminant, décomposition LU, ...), et nous obtiendrons souvent une expression simple de cette différentielle en fonction de l'entrée et de la sortie de la fonction considérée. En d'autres termes, si f est la fonction correspondant à l'opération considérée, f vérifiera souvent une équation différentielle de la forme $f' = g \circ (f, \text{id})$ où g est une fonction donnée, dont l'étude n'est, on l'espère, pas trop difficile. Le but de cette Sous-Section est d'étudier une telle équation différentielle et d'en déduire des bornes sur la fonction $\Lambda(f)$, et ainsi, sur le rayon à partir duquel la conclusion du Lemme 2.2.4 est satisfaite. *Nous supposons ici que K est de caractéristique nulle.*

Soit E, F et G des espaces vectoriels normés de dimension finie, avec $U \subseteq E$, $V \subseteq F$ et $W \subset G$ des ouverts. En généralisant le contexte précédent, nous nous intéressons à l'équation différentielle suivante :

$$f' = g \circ (f, h). \quad (2.10)$$

Ici $g : V \times W \rightarrow \text{Hom}(E, F)$ et $h : U \rightarrow W$ sont des fonctions localement analytiques données et $f : U \rightarrow V$ est la fonction localement analytique inconnue. Dans ce qui suit, nous supposons que V et W contiennent 0, que $f(0) = 0$, $h(0) = 0$ et $g(0) \neq 0$. Ces hypothèses n'entraînent pas de perte de généralité. En effet, nous pouvons, d'une part, toujours faire une translation sur f et h (et sur g en conséquence) de manière à ce qu'ils s'annulent tous en 0, et d'autre part, afin de pouvoir appliquer la Proposition 2.3.14 la dérivée $f'(0)$ doit être surjective, et donc *a fortiori* non-nulle.

Nous supposons que sont aussi données deux fonctions croissantes et convexes Λ_g et Λ_h telles que $\Lambda(g) \leq \Lambda_g$ et $\Lambda(h) \leq \Lambda_h$. Nous supposons de plus qu'il existe ν tel que Λ_g est constante sur l'intervalle $]-\infty, \nu]^3$. Nous introduisons les fonctions τ_ν et Λ_f définies par :

$$\begin{aligned} \tau_\nu(x) &= x && \text{si } x \leq \nu \\ &= +\infty && \text{sinon} \end{aligned}$$

$$\text{et } \Lambda_f(x) = \tau_\nu \circ (\text{id} + \Lambda_g \circ \Lambda_h)(x + \alpha)$$

où $\alpha \in \mathbb{R}$ satisfait $\|n!\| \geq e^{-\alpha n}$ pour tout n . Si p est la caractéristique du corps résiduel, une valeur possible pour α est $\alpha = -\frac{p}{p-1} \cdot \log \|p\|$ si $p > 0$ et $\alpha = 0$ si $p = 0$. La proposition suivante sera montrée dans la Sous-Section 2.3.3.

Proposition 2.3.15. *Nous avons $\Lambda(f) \leq \Lambda_f$.*

La Figure 2.1 illustre la Proposition 2.3.15. En trait bleu continu est représenté le graphe de la fonction Λ_f . Un calcul rapide montre que, sur un voisinage de $-\infty$, cette fonction est donnée par $\Lambda_f(x) = x + \alpha + \mu$ où μ est la valeur prise par Λ_g sur l'intervalle $]-\infty, \nu]$. La Proposition 2.3.15 énonce que le graphe de $\Lambda(f)$ est sous la ligne brisée en trait bleu continu. De plus, nous remarquons que le développement de Taylor de $f(z)$ en 0 commence par $g(0)z$. Ainsi, sur un voisinage de $-\infty$, nous avons $\Lambda(f)(x) = x + \log \|g(0)\|$. En utilisant la convexité, nous obtenons :

$$\Lambda(f)(x) \geq x + \log \|g(0)\|,$$

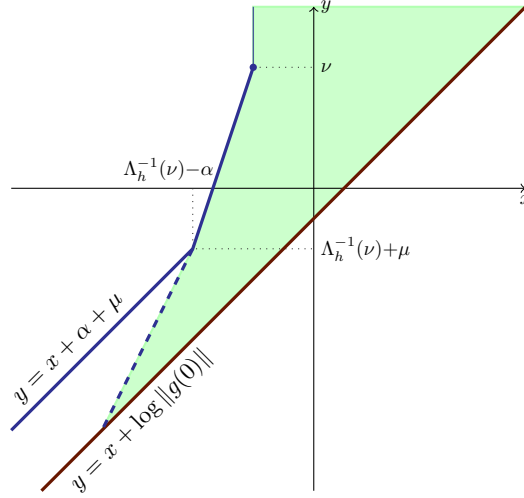
pour tout $x \in \mathbb{R}$. En d'autres termes, le graphe de $\Lambda(f)$ se situe au-dessus de la droite en marron. En outre, nous savons que les pentes de $\Lambda(f)$ sont toutes entières puisque f est localement analytique. Ainsi, $\Lambda(f)$ ne peut pas être au-dessus du trait bleu en pointillé défini par la droite de pente 2 passant par le première point de non-différentiabilité du trait bleu continu, qui est de coordonnées $(y_0 - \alpha - \mu, y_0)$ avec $y_0 = \min(\Lambda_h^{-1}(\nu) + \mu, \nu)$. En conclusion, nous avons montré que le graphe de $\Lambda(f)$ coïncide avec la droite en marron jusqu'à ce qu'il rencontre le trait en pointillé bleu, et ensuite, le graphe de $\Lambda(f)$ doit rester dans la région en vert.

Reprenons les notations de la Sous-Section 2.3.1 : si φ est une fonction convexe et $v \in \mathbb{R}$, nous avons défini

$$\varphi_{\geq v} : x \mapsto \inf_{y \geq 0} (\varphi(x + y) - vy).$$

Il s'agit de la plus grande fonction convexe telle que $\varphi_{\geq v} \leq \varphi$ et $\varphi'_{\geq v} \geq v$. Nous pouvons finalement en déduire la proposition suivante :

3. Nous pouvons remarquer que cette hypothèse est remplie si l'on prend $\Lambda_g = \Lambda(g)$ puisque nous avons supposé que $g(0)$ n'est pas nul.


 FIGURE 2.1. – Région admissible pour le graphe de $\Lambda(f)$

Proposition 2.3.16. *En conservant les notations précédentes, nous avons :*

$$\Lambda(f)_{\geq 2}(x) \leq 2(x + \alpha + \mu) - \min(\Lambda_h^{-1}(\nu) + \mu, \nu)$$

pour tout $x \leq \min(\Lambda_h^{-1}(\nu) - \alpha, \nu - \mu - \alpha)$.

Démonstration. L'inégalité est due au fait que $y = 2(x + \alpha + \mu) - y_0$ est l'équation de la droite en pointillé bleu. \square

C'est cette dernière proposition qui pourra par la suite être combinée à la Proposition 2.3.8 pour pouvoir appliquer le Lemme 2.2.4.

Remarque 2.3.17. Dans certains cas, il est possible que la fonction f étudiée soit solution d'une équation différentielle plus simple, de la forme $f' = g \circ f$. Si c'est le cas, la Proposition 2.3.16 nous donne la majoration suivante $\Lambda(f)_{\geq 2}(x) \leq 2(x + \alpha + \mu) - \nu$ pour $x \leq \nu - \mu - \alpha$.

Nous pouvons aussi modifier l'équation précédente en choisissant une norme sur F particulière $\|x\|'_F = e^\mu \cdot \|x\|_F$ ($x \in F$) et en prenant $h : (F, \|\cdot\|_F) \rightarrow (F, \|\cdot\|'_F)$ qui est l'identité de F . La fonction $\Lambda(h) : \mathbb{R} \rightarrow \mathbb{R}$ envoie alors x sur $x + \mu$ et nous pouvons choisir $\Lambda_h = \Lambda(h)$.

2.3.3. Démonstration de la Proposition 2.3.15

Dans cette Sous-Section, nous montrons la Proposition 2.3.15 dans le contexte, plus général, des K -espaces de Banach.

Composée de fonctions localement analytiques

Soit U, V et W trois ouverts des K -espaces de Banach E, F et G , respectivement. Nous supposons que $0 \in U, 0 \in V$. Soit $f : U \rightarrow V$ et $g : V \rightarrow W$ deux fonctions localement analytiques au voisinage de 0, avec $f(0) = 0$. La composition $h = g \circ f$ est alors elle aussi localement analytique au voisinage de 0. Soit $f = \sum_{n \geq 0} f_n, g = \sum_{n \geq 0} g_n$ et $h = \sum_{n \geq 0} h_n$ les développements analytiques de f, g et h . Ici f_n, g_n et h_n sont les restrictions à la diagonale de formes n -linéaires symétriques F_n, G_n et H_n , respectivement. Le but de ce paragraphe est de montrer le résultat intermédiaire suivant :

Proposition 2.3.18. *Avec les notations précédentes, nous avons*

$$\|h_r\| \leq \sup_{m, (n_i)} \|g_m\| \cdot \|f_{n_1}\| \cdots \|f_{n_m}\|$$

pour tout entier positif r , où le sup est pris sur tous les couples $(m, (n_i))$ où m est un entier positif et $(n_i)_{1 \leq i \leq m}$ est une suite de longueur m d'entiers positifs tels que $n_1 + \dots + n_m = r$.

2. Le lemme de précision

Nous pouvons développer $g \circ f$ pour obtenir :

$$\sum \binom{m}{k_1 \dots k_\ell} G_m(f_{n_1}, \dots, f_{n_1}, \dots, f_{n_\ell}, \dots, f_{n_\ell}) \quad (2.11)$$

où $\binom{m}{k_1 \dots k_\ell} = \frac{m!}{k_1! \dots k_\ell!}$ est le coefficient multinomial et la somme parcourt :

- (a) toutes les suites finies (k_i) d'entiers positifs de longueur (resp. somme) notée ℓ (resp. m), et
- (b) toutes les suites finies (n_i) d'entiers positifs de longueur ℓ .

En outre, en argument de G_m , la variable f_{n_i} est répétée k_i fois.

Le degré de $G_m(f_{n_1}, \dots, f_{n_1}, \dots, f_{n_\ell}, \dots, f_{n_\ell})$ est $r = k_1 n_1 + \dots + k_\ell n_\ell$ et ainsi contribue à h_r . En conséquence, h_r est égal à (2.11) où la somme est restreinte aux suites (k_i) , (n_i) telles que $k_1 n_1 + \dots + k_\ell n_\ell = r$. La Proposition 2.3.18 est alors conséquence du lemme suivant :

Lemme 2.3.19. *Soit E un K -espace vectoriel normé. Soit $\varphi : E^m \rightarrow K$ une forme m -linéaire symétrique et $\psi : E \rightarrow K$ définie par $\psi(x) = \varphi(x, x, \dots, x)$. Étant donné des entiers positifs k_1, \dots, k_ℓ dont la somme est m et $x_1, \dots, x_\ell \in E$, nous avons*

$$\left\| \binom{m}{k_1 \dots k_\ell} \cdot \varphi(x_1, \dots, x_1, \dots, x_\ell, \dots, x_\ell) \right\| \leq \|\psi\| \cdot \|x_1\|^{k_1} \dots \|x_\ell\|^{k_\ell}$$

où, dans le membre de gauche, la variable x_i est répétée k_i fois.

Démonstration. Il suffit de montrer que

$$\left\| \binom{m}{k_1 \dots k_\ell} \cdot \varphi(x_1, \dots, x_1, \dots, x_\ell, \dots, x_\ell) \right\| \leq \|\psi\|$$

en supposant que les x_i sont de norme au plus 1. Nous procédons par récurrence sur ℓ . Le cas $\ell = 1$ est une conséquence directe de la définition de $\|\psi\|$. Prenons maintenant $(\ell + 1)$ entiers $k_1, \dots, k_{\ell+1}$ dont la somme est m , et $(\ell + 1)$ éléments $x_1, \dots, x_{\ell+1}$ de la boule unité de E . Introduisons une nouvelle variable λ , prise dans \mathcal{O}_K . Nous posons $x'_i = x_i$, $k'_i = k_i$ où $i < \ell$ et $x'_\ell = x_\ell + \lambda x_{\ell+1}$ et $k'_\ell = k_\ell + k_{\ell+1}$. Par hypothèse de récurrence, nous savons que l'inégalité

$$\left\| \binom{m}{k'_1 \dots k'_\ell} \cdot \varphi(x'_1, \dots, x'_1, \dots, x'_\ell, \dots, x'_\ell) \right\| \leq \|\psi\|$$

est vraie pour $\lambda \in K$. En outre, le membre de gauche de l'inégalité est un polynôme $P(\lambda)$ de degré k'_ℓ dont le coefficient en λ^j est

$$\binom{m}{k'_1 \dots k'_\ell} \cdot \binom{k'_\ell}{j} \cdot \varphi(\underline{x}_j) = \binom{m}{k_1 \dots k_{\ell-1} j} \cdot \varphi(\underline{x}_j)$$

avec $\underline{x}_j = (x_1, \dots, x_1, \dots, x_{\ell+1}, \dots, x_{\ell+1})$ où x_i est répété k_i fois si $i < \ell$ et x_ℓ (resp. $x_{\ell+1}$) est répété j fois (resp. $k'_\ell - j$ fois). Puisque $\|P(\lambda)\| \leq \|\psi\|$ pour tout λ dans la boule unité, la norme de chacun de ses coefficients est aussi au plus $\|\psi\|$. En regardant le coefficient de λ^{k_ℓ} , le résultat est obtenu. \square

Majoration de $\Lambda(f)$

Nous revenons au cadre de la Proposition 2.3.15. Soit $f = \sum_{n \geq 0} f_n$, $g = \sum_{n \geq 0} g_n$ et $h = \sum_{n \geq 0} h_n$ le développement analytique de f , g et h . Ici f_n , g_n et h_n sont la restriction à la diagonale de formes n -linéaires symétriques F_n , G_n et H_n , respectivement. Nous rappelons que $\Lambda(f)$ est la transformée de Legendre du polygone de Newton de $\text{NP}(f)$ défini en Sous-Section 2.3.1, et que α est un nombre réel tel que $\|n!\| \geq e^{-\alpha n}$ pour tout $n \in \mathbb{N}$.

Lemme 2.3.20. *Avec les notations précédentes, si (a, b) satisfait $b \geq a + \Lambda(g)(\max(b, \Lambda(h)(a)))$ alors $b \geq \Lambda(f)(a - \alpha)$.*

Démonstration. Nous avons $f' = \sum_{n \geq 0} f'_n$ où

$$f'_n : U \rightarrow \mathcal{L}(E, F), \quad x \mapsto (h \mapsto n \cdot F_n(h, x, x, \dots, x)).$$

En prenant $h = x$, nous trouvons $\|f'_n\| \geq \|n f_n\| = \|n\| \cdot \|f_n\|$. En combinant ce fait avec la Proposition 2.3.18, nous obtenons

$$\|(r+1)f_{r+1}\| \leq \sup_{m, (n_i)} \|g_m\| \cdot \prod_{i=1}^m \max(\|f_{n_i}\|, \|h_{n_i}\|)$$

pour tout entier positif r , où le sup est pris sur tous les couples $(m, (n_i))$ où m est un entier positif et $(n_i)_{1 \leq i \leq m}$ une suite de longueur m d'entiers positifs tels que $n_1 + \dots + n_m = r$. Nous posons $u_r = \|r! f_r\|$. En multipliant l'inégalité précédente par $\|r!\|$, nous obtenons :

$$u_{r+1} \leq \sup_{m, (n_i)} \|g_m\| \cdot \prod_{i=1}^m \max(u_{n_i}, \|n_i! h_{n_i}\|) \quad (2.12)$$

puisque le coefficient multinomial $\binom{r}{n_1 \dots n_m}$ est un entier, et donc, de norme au plus 1. Soit maintenant deux réels a et b satisfaisant l'hypothèse du lemme. Soit $d = \Lambda(h)(a)$. En utilisant la définition de $\Lambda(h)$ et de la transformée de Legendre, nous obtenons $\|h_n\| \leq e^{-an+d}$ pour tout n . De la même manière, avec l'hypothèse faite sur (a, b) , nous obtenons $\|g_m\| \leq e^{-\max(b, d) \cdot m + b - a}$ pour tout m . Nous pouvons maintenant prouver que $u_r \leq e^{-ar+b}$ par récurrence sur r . Lorsque $r = 0$, le résultat est évident puisque u_0 s'annule. Autrement, la récurrence est conséquence de ce qui suit :

$$\begin{aligned} u_{r+1} &\leq \sup_{m, (n_i)} e^{-\max(b, d) \cdot m + b - a + \sum_{i=1}^m (-an_i + \max(b, d))} \\ &= e^{b-a-ar} = e^{-a(r+1)+b}. \end{aligned}$$

De par la définition de u_r , nous obtenons $\|f_r\| \leq u_r \cdot \|r!\|^{-1} \leq e^{-(a-\alpha)r+b}$. Ainsi $b \geq \Lambda(f)(a-\alpha)$. \square

Nous pouvons maintenant conclure la preuve de la Proposition 2.3.15 comme suit. Soit $a \in \mathbb{R}$ et $b = a + \Lambda_g \circ \Lambda_h(a)$, et prouvons que $\Lambda(f)(a-\alpha) \leq b$ lorsque $b \leq \nu$. Grâce à la Proposition 2.3.15, il est suffisant de vérifier qu'un tel couple (a, b) satisfait l'hypothèse du Lemme 2.3.20. Clairement, $b \geq a + \Lambda(g) \circ \Lambda(h)(a)$ puisque $\Lambda_g \geq \Lambda(g)$, $\Lambda_h \geq \Lambda(h)$ et Λ_g est croissante. En outre, puisque $b \leq \nu$, nous obtenons $\Lambda_g(b) = \min_{x \in \mathbb{R}} \Lambda_g(x) \leq \Lambda_g \circ \Lambda_h(a)$, dont nous déduisons $a + \Lambda_g(b) \leq a + \Lambda_g \circ \Lambda_h(a) = b$.

2.4. Premières idées de mise en œuvre en pratique

Considérons une fonction \mathbf{f} qui prend en entrée un élément approché dans un ouvert U d'un K -espace de Banach E et renvoie en sortie un autre élément approché dans un ouvert V d'un autre K -espace de Banach F . Dans les applications, cette fonction modèle une fonction mathématique continue $f : U \rightarrow V$: si \mathbf{f} est appelée avec en entrée $x + O(H)$, elle retourne $x' + O(H')$ avec $f(x+H) \subseteq x' + H'$. Nous disons que \mathbf{f} *préserve la précision* si l'inclusion précédente est une égalité. Comme nous l'avons vu dans la Section 1.4, c'est loin d'être toujours le cas.

Supposons maintenant que f est localement analytique sur U et que $f'(x)$ est surjective. La Proposition 2.3.8 nous donne alors une condition suffisante simple pour décider si un réseau donné H est un réseau du premier ordre pour f en x . Pour un tel réseau, par définition, nous avons $f(x+H) = f(x) + f'(x)(H)$ et ainsi \mathbf{f} doit retourner $f(x) + O(f'(x)(H))$ si elle préserve la précision. Dans cette section, nous étudions comment, sous les hypothèses précédentes, travailler avec \mathbf{f} de manière à ce qu'elle retourne toujours une sortie avec la précision optimale.

2.4.1. Calcul en un passage

L'exécution de la fonction \mathbf{f} engendre une factorisation :

$$f = f_n \circ f_{n-1} \circ \dots \circ f_1$$

2. Le lemme de précision

avec les f_i correspondant à chaque étape élémentaire du calcul (*e.g.* addition, multiplication ou création de nouvelles variables). Ces étapes sont toutes localement analytiques. Pour tout i , soit U_i le codomaine de f_i . Bien sûr, U_i doit contenir toutes les valeurs possibles pour toutes les valeurs des variables qui sont définies dans le programme après exécution de la i -ème étape. Mathématiquement, nous supposons que U_i est un ouvert dans un K -espace de Banach E_i . Nous avons $U_n = V$ et le domaine de définition de f_i est U_{i-1} avec par convention, $U_0 = U$. Pour tout i , nous posons $g_i = f_i \circ \dots \circ f_1$ et $x_i = g_i(x)$.

Lorsque nous exécutons la fonction \mathbf{f} sur l'entrée $x + O(H)$, nous appliquons d'abord f_1 à cette entrée, pour obtenir de cette manière un premier résultat $x_1 + O(H_1)$ et ensuite continuons avec f_2, \dots, f_n . À chaque étape, nous obtenons un résultat intermédiaire que nous noterons par $x_i + O(H_i)$. Une manière de garantir que la précision est préservée est de vérifier que $H_i = f'_i(x)(H_{i-1}) = g'_i(x)(H)$ à chaque étape. Ceci peut être réalisé en calculant avec les f'_i en même temps qu'avec les f_i et en appliquant f'_i à H_{i-1} .

Il y a néanmoins un problème avec cette approche : afin de pouvoir appliquer le Lemme 2.2.4, nous devons connaître *a priori* les valeurs exactes de tous les x_i pour connaître les différentielles en ces points, et ce n'est bien sûr pas possible ! En supposant que $g'_i(x)$ est surjective pour tout i , nous pouvons procéder comme suit. Pour tout i , nous fixons un réseau du premier ordre \tilde{H}_i pour g_i en x . Sous ces hypothèses, de tels réseaux existent toujours et peuvent être calculés grâce à la Proposition 2.3.8 et au Lemme 2.3.3 (voir aussi la Remarque 2.3.4). Maintenant, l'égalité $g_i(x + \tilde{H}_i) = x_i + g'_i(x)(\tilde{H}_i)$ signifie que toute perturbation de x_i par un élément dans $g'_i(x)(\tilde{H}_i)$ est induit par une perturbation de x par un élément dans $\tilde{H}_i \subset H$. Ainsi, nous pouvons librement calculer x_i modulo $g'_i(x)(\tilde{H}_i)$ sans modifier le résultat final. Puisque $g'_i(x)(\tilde{H}_i)$ est un réseau de E_i , cette remarque rends possible le calcul de x_i .

Remarque 2.4.1. Dans certains cas, il est en fait possible de déterminer par des arguments mathématiques des réseau \tilde{H}_i convenables, ainsi que leurs images par $g'_i(x)$ (ou, au moins, de bonnes approximations de celles-ci) avant de débiter le calcul. Si possible, ceci peut beaucoup aider. Nous verrons aux Chapitres 4 et 5 une méthode pour travailler dans ce cadre, puis des exemples (suite de Somos-4, certaines équations différentielles à séparation des variables).

2.4.2. Calcul en deux passages

L'approche précédente fonctionne seulement si les $g'_i(x)$ sont toutes surjectives. Malheureusement, cette hypothèse n'est en général pas satisfaite. En effet, la dimension de E_i correspond essentiellement au nombre de variables utilisées à l'étape i . Si toutes les $g'_i(x)$ étaient surjectives, cela signifierait que la fonction \mathbf{f} n'initialise jamais de nouvelle variable ! Dans ce qui suit, nous proposons une autre solution qui ne suppose pas la surjectivité des $g'_i(x)$.

Pour $i \in \{1, \dots, n\}$, nous définissons $h_i = f_n \circ \dots \circ f_{i+1}$, de manière à avoir $f = h_i \circ g_i$. Sur les différentielles, nous avons $f'(x) = h'_i(x_i) \circ g'_i(x)$. Puisque $f'(x)$ est surjective (par hypothèse), nous en déduisons que $h'_i(x_i)$ est surjective pour tout i . Soit H'_i un réseau de E_i tel que :

- (a) H'_i est contenu dans $H_i + \ker h'_i(x_i) = h'_i(x_i)^{-1}(f'(x)(H))$;
- (b) H'_i est un réseau du premier ordre pour h_i en x_i .

Par définition, nous avons $h_i(x_i + H'_i) = x_n + h'_i(x_i)(H'_i) \subset x_n + f'(x)(H)$. En conséquence, si l'on modifie les valeurs intermédiaires x_i par un élément de H'_i , alors à la i -ème étape d'exécution de \mathbf{f} , le résultat reste inchangé. En d'autres mots, il est suffisant de calculer x_i modulo H'_i .

Il n'est néanmoins pas évident de mettre ces idées en pratique puisque lorsque l'on entre dans la i -ème étape de l'exécution de \mathbf{f} , nous n'avons pas encore calculé h_i , et ainsi, nous ne sommes *a priori* pas capable de déterminer un réseau H'_i satisfaisant aux hypothèses (a) et (b) précédentes. Une solution possible à ce problème est de précéder en plusieurs étapes de la manière suivante :

- (1) Pour i de 1 à n , nous calculons $x_i, f'_i(x_{i-1})$ à une précision petite (mais suffisante pour la seconde étape) en même temps qu'une borne supérieure pour la fonction $\Lambda(h \mapsto f_i(x_{i-1} + h) - f_i(x_{i-1}))$;
- (2) pour i de n à 1, nous calculons $h'_i(x_i)$ et déterminons un réseau H'_i satisfaisant (a) et (b) ;
- (3) pour i de 1 à n , nous calculons x_i modulo H'_i et finalement renvoyons en sortie $x_n + O(f'(x)(H))$.

Si l'on utilise l'algorithmique détendue pour les calculs sur les éléments de K (*cf* [vdH02], [vdH07], [BvdHL11]), nous pouvons réutiliser à l'Étape (3) les calculs déjà effectués lors de l'Étape (1). La

méthode de calcul en deux passages que nous venons de présenter n'est alors probablement pas beaucoup plus coûteuse que la méthode en un passage, mais est cependant plus difficile à mettre en pratique.

Nous concluons cette section en remarquant que la méthode en deux passages semble particulièrement bien adaptée à des calculs paresseux ou détendus sur les p -adiques. En effet, dans ce contexte, une précision en sortie est fixée et c'est l'implémentation qui détermine automatiquement la précision nécessaire sur les entrées pour atteindre cette précision. Pour cela, elle calcule d'abord le squelette du calcul (*i.e.* elle détermine les fonctions f_i et éventuellement les x_i à une petite précision suffisante lorsque des points de branchement apparaissent et qu'il faut choisir quelle branche suivre) et ensuite travaille avec ce squelette en remontant vers les entrées pour déterminer (une majoration) de la précision nécessaire à chaque étape.

2.4.3. Remarques sur la surjectivité

Depuis le début, nous avons supposé que $f'(x)$ est surjective. Nous discutons ici de ce qui peut arriver lorsque cette hypothèse est relâchée. Comme il est expliqué après la preuve du Lemme 2.2.4, la première chose que nous pouvons faire est de projeter sur différents quotients, *i.e.* travailler avec des compositions $\text{pr}_W \circ f$ pour une famille assez large de sous-espaces vectoriels fermés $W \subset F$ tels que $W + f'(x)(E) = F$. Si F a un système de coordonnées naturel, nous pouvons généralement prendre les pr_W comme les projections sur chaque coordonnée. En faisant ainsi, nous obtenons une précision sur chaque coordonnée, individuellement. En outre, nous avons la garantie que la précision sur chaque coordonnée est optimale, et ce même si le réseau diagonal que l'on peut construire à partir d'elles ne l'est pas. Ceci revient à projeter sur le type de précision diagonal.

Illustrons la discussion précédente sur un exemple : supposons que nous voulons calculer avec la fonction $f : (K^n)^n \rightarrow M_n(K)$ qui envoie une famille de n vecteurs sur leur matrice de Gram. La différentielle de f n'est clairement jamais surjective puisque f prends ses valeurs dans le sous-espace constitué des matrices symétriques. Néanmoins, pour toutes paires $(i, j) \in \{1, \dots, n\}^2$, on peut considérer la composition $f_{ij} = \text{pr}_{ij} \circ f$ où $\text{pr}_{ij} : M_n(K) \rightarrow K$ envoie une matrice M sur sa (i, j) -ème entrée. Les applications f_{ij} sont différentiables et leurs différentielles sont surjectives en générales (dès qu'elles sont non nulles). Soit M une matrice connue à une précision finie donnée, telle que $f'_{ij}(M) \neq 0$ pour tout (i, j) . Nous pouvons alors appliquer un calcul en un ou deux passages et obtenir $f_{ij}(M)$ ainsi que sa précision. Au total, nous pouvons ainsi reconstituer la matrice complète $f(M)$ ainsi qu'une donnée de précision optimale sur chaque coordonnée.

L'étude de cet exemple suggère par ailleurs une autre solution pour le problème de la non surjectivité. En effet, nous remarquons que la fonction f donnée en exemple n'a pas une différentielle surjective parce que son codomaine est trop gros. En le remplaçant par le K -espace vectoriel $S_n(K)$ des matrices symétriques $n \times n$ sur K , nous rendons f' surjective. Même si l'image par une application $f : E \rightarrow F$ est rarement un sous-espace vectoriel de F , elle est souvent une sous- K -variété de F . Nous pouvons alors appliquer les résultats de la Section 2.5 pour étudier $f : U \rightarrow f(U)$, qui a de meilleures chances d'avoir une différentielle surjective.

2.5. Généralisation aux variétés

Beaucoup d'objets p -adiques naturels ne vivent pas dans des espaces vectoriels : des points dans un espace projectif ou sur une courbe elliptique, des sous-espaces vectoriels d'un espace vectoriel donné (qui vivent dans une grassmannienne), des classes d'isomorphismes de certaines courbes (qu'on peut voir vivre dans divers espaces de modules), ... Dans cette Section, nous étendons le formalisme développé dans la Section 2.2 à un cadre plus général : nous considérons le cas des variétés différentiables localement modelées sur des K -espaces de Banach ultramétriques. Ce cadre recouvre bien tous les exemples que nous venons de citer.

2.5.1. K -variétés différentiables

La théorie des K -variétés différentiables de dimension finie est, par exemple, présentée dans [Sch11] Ch. 8-9. Dans cette Section, nous travaillerons avec une notion légèrement différente de variété,

2. Le lemme de précision

qui permet de manipuler des K -espaces de Banach possiblement de dimension infinie. Comme espaces de Banach de dimension infinie, nous pouvons citer certains espaces de séries convergentes ou sur-convergentes. Il peut être particulièrement agréable de pouvoir travailler dans ce cadre si l'on s'intéresse aux géométries rigides ou non-archimédiennes. Précisons notre définition : pour nous, une K -variété différentiable (ou simplement K -variété pour simplifier) est la donnée d'un espace topologique V avec un recouvrement ouvert $V = \bigcup_{i \in I} V_i$ (où I est un ensemble donné) et, pour tout $i \in I$, un homéomorphisme $\varphi_i : V_i \rightarrow U_i$ où U_i est un ouvert d'un K -espace de Banach E_i tel que pour tout $i, j \in I$ pour lesquels $V_i \cap V_j$ est non-vide l'application composée

$$\psi_{ij} : \varphi_i(V_{ij}) \xrightarrow{\varphi_i^{-1}} V_{ij} \xrightarrow{\varphi_j} \varphi_j(V_{ij}) \quad (\text{avec } V_{ij} = V_i \cap V_j) \quad (2.13)$$

est différentiable. Nous rappelons que les applications φ_i ci-dessus sont appelés des *cartes*. Les ψ_{ij} sont des applications de transition. La collection des φ_i et des ψ_{ij} est appelée un *atlas* de V . Dans la suite, nous supposons de plus que le recouvrement ouvert $V = \bigcup_{i \in I} V_i$ est localement fini, ce qui signifie que chaque point $x \in V$ appartient à un nombre fini de V_i . Un exemple trivial de K -variété est un K -espace de Banach.

Si V est une K -variété et x un point de V , nous définissons l'espace tangent $T_x V$ de V en x comme l'espace E_i pour un certain i tel que $x \in V_i$. Nous remarquons que si x est à la fois dans V_i et V_j , l'application linéaire $\psi'_{ij}(\varphi_i(x))$ définit un isomorphisme entre E_i et E_j . En outre ces isomorphismes sont compatibles de manière évidente. Ceci implique que la définition de $T_x V$ donnée plus haut ne dépend pas (à un isomorphisme canonique près) de l'indice i tel que $x \in V_i$ et est ainsi bien définie.

Comme on le fait habituellement, nous pouvons définir la notion de différentiabilité (en un point) pour une application continue entre deux K -variétés en passant par les cartes. Une application différentiable $f : V \rightarrow V'$ induit une application linéaire sur l'espace tangent $f'(x) : T_x V \rightarrow T_{f(x)} V'$ pour tout x dans V . Elle est appelée la *différentielle* de f en x .

2.5.2. Données de précision

Définition

Revenons à notre problème de précision. Étant donné V une K -variété comme défini plus haut, nous aimerions pouvoir donner du sens à ce que nous appellerions des éléments approchés de V à une précision donnée, qui seraient des expressions de la forme $x + O(H)$ où x est dans une partie dense calculable de V et H est une *donnée de précision*. Pour cela, fixons une K -variété V . Nous utiliserons librement les notations I, V_i, φ_i, \dots introduites dans la Sous-Section 2.5.1.

Définition 2.5.1. Soit $x \in V$. Une *donnée de précision* en x est un réseau dans l'espace tangent $T_x V$ tel que pour tout indices i et j avec $x \in U_i \cap U_j$, l'image de $T_x V$ dans E_i est un réseau du premier ordre pour ψ_{ij} en $\varphi_i(x)$ (cf Définition 2.2.3).

Remarque 2.5.2. La définition de donnée de précision en x dépend non seulement de x et de la variété V où il se trouve, mais aussi de l'atlas choisi sur V .

Heureusement, nous montrons que cette définition ne dépend pas de la carte :

Lemme 2.5.3. Soit $x \in V$ et H une donnée de précision en x . Le sous-ensemble

$$\varphi_i^{-1}(\varphi_i(x) + \varphi'_i(x)(H)) \subset V$$

ne dépend pas de l'indice i tel que $x \in V_i$.

Démonstration. Soit i et j deux indices tels que x appartienne à V_i et V_j . Soit $x_i = \varphi_i(x) \in E_i$ et $H_i = \varphi'_i(x)(H)$. L'égalité

$$\varphi_i^{-1}(\varphi_i(x) + \varphi'_i(x)(H)) = \varphi_j^{-1}(\varphi_j(x) + \varphi'_j(x)(H))$$

est clairement équivalente à $\psi_{ij}(x_i + H_i) = \psi_{ij}(x_i) + \psi'_{ij}(x_i)(H_i)$ et la dernière est vraie puisque H_i est un réseau du premier ordre pour ψ_{ij} en x_i . \square

Nous pouvons ainsi définir $x + O(H)$:

Définition 2.5.4. Soit $x \in V$ et H une donnée de précision en x . Nous posons :

$$x + O(H) = \varphi_i^{-1}(\varphi_i(x) + \varphi'_i(x)(H)) \subset V$$

pour un (ou de manière équivalente, pour tout) i tel que $x \in V_i$.

Changement de point de base

Afin de pouvoir nous restreindre à regarder des x vivant dans une partie calculable dense de V , nous avons besoin de comparer $x_0 + O(H_0)$ et les $x + O(H)$ lorsque x et x_0 sont assez proches. Regardons d'abord ce qu'il advient lorsque nous avons une carte fixée donnée : nous choisissons un $i \in I$ et prenons deux éléments x_0 et x dans V_i . Nous considérons de plus un réseau \tilde{H}_0 dans E_i (que nous pouvons penser être $\varphi'_i(x_0)(H_0)$) et nous souhaitons construire un réseau \tilde{H} tel que $\varphi_i(x_0) + \tilde{H}_0 = \varphi_i(x) + \tilde{H}$. Bien sûr, $\tilde{H} = \tilde{H}_0$ convient dès que $\varphi_i(x) - \varphi_i(x_0) \in \tilde{H}_0$. Maintenant, remarquons que les espaces tangents $T_{x_0}V$ et T_xV sont tous deux isomorphes à E_i via les applications $\varphi'_i(x_0)$ et $\varphi'_i(x)$ respectivement. Un choix naturel de candidat H est alors :

$$H = (\varphi'_i(x)^{-1} \circ \varphi'_i(x_0))(H_0). \quad (2.14)$$

Avec ce choix, $x + O(H) = x_0 + O(H_0)$ pourvu que x et x_0 soient assez proches au sens suivant : la différence $\varphi_i(x) - \varphi_i(x_0)$ vit dans le réseau $\varphi'_i(x_0)(H_0)$. Nous avons de plus indépendance par rapport à i .

Proposition 2.5.5. Soit $x_0 \in V$ et H_0 une donnée de précision en x_0 . Alors, pour tout x suffisamment proche de x_0 ,

- (i) le réseau H défini par (2.14) ne dépend pas de i et est une donnée de précision en x , et
- (ii) nous avons $x + O(H) = x_0 + O(H_0)$.

Démonstration. Nous prouvons d'abord (i). Pour un indice i tel que $x, x_0 \in V_i$, notons par $f_i : T_{x_0}V \rightarrow T_xV$ la composée $\varphi'_i(x)^{-1} \circ \varphi'_i(x_0)$. Étant donné un indice j satisfaisant la même hypothèse, la différence $f_i - f_j$ tend vers 0 lorsque x tend vers x_0 (voir la Remarque 2.2.2). Puisque H_0 est ouvert dans $T_{x_0}V$, ceci implique que $(f_j - f_i)(H_0)$ contient $f_i(H_0)$ et $f_j(H_0)$ si x et x_0 sont assez proches. Maintenant, prenons $w \in f_j(H_0)$ et écrivons $w = f_j(v)$ avec $v \in H_0$. Alors w est égal à $f_i(v) + (f_j - f_i)(v)$ et ainsi appartient à $f_i(H_0)$ puisque c'est le cas de chaque terme de la somme. Ainsi, $f_j(H_0) \subset f_i(H_0)$. L'inclusion réciproque est prouvée de la même manière. Le fait que H est une donnée de précision en x est facile et laissé au lecteur. Finalement, si x est assez proche de x_0 , il suffit de vérifier (ii) dans les cartes, mais cela a déjà été fait. \square

2.5.3. Généralisation du lemme principal

Avec les définitions précédentes, le Lemme 2.2.4 s'étend aux variétés. Pour le montrer, nous devons d'abord définir une norme sur l'espace tangent T_xV (où V est une K -variété donnée et x un point de V). Il n'y a en général pas de choix canonique pour cela. En effet, considérons une K -variété V recouverte par des cartes U_i ($i \in I$) qui sont des ouverts de K -espaces de Banach E_i . Si x est un point de V , l'espace tangent T_xV est par définition isomorphe à E_i pour chaque indice i tel que $x \in V_i$. Une norme naturelle sur T_xV serait alors celle obtenue par tiré en arrière de la norme sur E_i . Cependant, comme les applications de transition ne sont pas supposées être des isométries, cette norme dépend du choix de i . Elles sont néanmoins toutes équivalentes puisque les applications de transition sont supposées être continues.

Pour le lemme suivant, nous choisissons n'importe laquelle des normes proposées ci-dessus pour T_xV .

Lemme 2.5.6. Soit V et W deux K -variétés. Supposons que nous disposons d'une application différentiable $f : V \rightarrow W$, et d'un point $x \in V$ tel que $f'(x) : T_xV \rightarrow T_{f(x)}W$ est surjective.

Alors, pour tout $\rho \in]0, 1]$, il existe un réel $\delta \in \mathbb{R}_{>0}$ tel que, pour tout $r \in]0, \delta]$, tout réseau H dans T_xV tel que $B_{T_xV}^-(\rho r) \subset H \subset B_{T_xV}(r)$ est un réseau du premier ordre pour f en x .

2. Le lemme de précision

Démonstration. Appliquer le Lemme 2.2.4 dans les cartes. \square

Remarque 2.5.7. La constante δ qui apparaît dans le lemme dépend (à une constante multiplicative près) de la norme choisie sur $T_x V$. Cependant, une fois cette norme choisie, et en supposant de plus que V et W sont des K -variétés localement analytiques et que l'application f est localement analytique elle aussi, la constante δ peut être rendue explicite en utilisant les méthodes de la Section Sous-Section 2.3.1.

2.5.4. Un premier exemple

Nous illustrons la théorie développée dans cette Section avec l'exemple des courbes elliptiques. D'autres exemples suivront au Chapitre 3, en particulier concernant les grassmanniennes, en Section 3.3.

Dans cet exemple, pour raison de simplicité, nous supposons que K n'est pas de caractéristique 2. Soit a et b deux éléments de K tels que $4a^3 + 27b^2 \neq 0$ et soit E le sous-ensemble de K^2 composé des couples (x, y) satisfaisant l'équation $y^2 = x^3 + ax + b$. Soit $\text{pr}_x : E \rightarrow K$ (resp. $\text{pr}_y : E \rightarrow K$) l'application qui envoie un couple (x, y) sur x (resp. sur y).

Nous supposons d'abord que a et b vivent dans le sous-anneau R des éléments exacts. Pour chaque point $P_0 = (x_0, y_0)$ de E excepté peut-être un nombre fini d'entre eux, l'application pr_x définit un difféomorphisme d'un ouvert contenant P_0 vers un ouvert de K ; la même chose est vraie pour pr_y . De plus, autour de chaque $P_0 \in E$, au moins l'une de ces projections satisfait la condition précédente. Ainsi, les applications pr_x et pr_y définissent ensemble un atlas de E , lui conférant une structure de K -variété.

Soit P_0 un point dans E autour duquel pr_x et pr_y définissent tous deux des cartes. Le Lemme 2.5.3 nous dit alors qu'une donnée de précision sur x détermine une donnée de précision sur y et réciproquement. En effet, dans un voisinage de P_0 nous pouvons écrire $y = \sqrt{x^3 + ax + b}$ (pour un certain choix de racine carrée) et obtenir la précision sur y à partir de la précision sur x en appliquant le Lemme 2.2.4. Nous pouvons aller dans l'autre direction de la même manière en écrivant x localement comme une fonction de y . Une donnée de précision en P_0 n'est alors qu'une donnée de précision sur la coordonnée x ou sur la coordonnée y , en gardant à l'esprit que chacune détermine l'autre. Voir une donnée de précision en P_0 comme un réseau dans l'espace tangent est une bonne manière de rendre ce choix canonique, mais en pratique, il suffit de choisir une des coordonnées et de suivre la précision sur celle-ci.

Nous concluons l'étude de cet exemple en présentant une méthode simple pour transformer une donnée de précision sur x en une donnée de précision sur y et réciproquement. En différentiant l'équation de la courbe elliptique, nous obtenons :

$$2y \cdot dy = (3x^2 + a) \cdot dx. \quad (2.15)$$

Dans l'équation ci-dessus, dx et dy peuvent être pensées comme de petites perturbations de x et y respectivement. L'équation (2.15) donne alors une relation linéaire entre la précision sur x (qui est représentée par dx) et celle sur y (représentée par dy). Cette relation correspond exactement à celle donnée par le Lemme 2.2.4.

Finalement, considérons le cas où a et b sont eux-même connus à précision finie et E n'est pas complètement déterminée. Nous ne pouvons alors pas la voir directement comme une K -variété et la discussion précédente ne s'applique pas directement. Néanmoins, nous pouvons toujours considérer la sous-variété de K^4 donnée par les points (a, b, x, y) satisfaisant $y^2 = x^3 + ax + b$. Les projections sur les hyperplans $a = 0$, $b = 0$, $x = 0$ et $y = 0$ respectivement définissent des cartes pour cette K -variété. À partir de là, nous pouvons voir qu'une donnée de précision sur une courbe elliptique non-définie à précision infinie E est une donnée de précision sur un triplet de trois variables parmi a , b , x et y .

3. Applications du lemme, calcul de différentielles

"For great justice."

Captain, *Zero Wing*

"After all, it could only cost you your life, and you got that for free!"

Earthbound

Le but du présent chapitre est de fournir dans des cadres divers des exemples d'applications du formalisme développé au chapitre précédent. Il s'agit d'un travail en commun avec Xavier Caruso et David Roe. Autant que possible, nous nous attacherons à exposer les conséquences pratiques de ces applications. Nous débutons avec l'étude en Section 3.1 des applications de la précision différentielle aux calculs classiques sur les polynômes : pgcd, factorisation, évaluation, ...

Ensuite, nous nous intéressons à l'algèbre linéaire, d'abord en Section 3.2 où nous étudions les calculs matriciels tels que la multiplication ou le calcul d'une factorisation LU. Nous clôturons ce chapitre avec en Section 3.3 une présentation du calcul, à précision finie, sur les grassmanniennes, et de ses conséquences lorsqu'on s'intéresse au calcul de l'intersection ou de la somme d'espaces vectoriels.

3.1. Polynômes

Pour tout entier d , notons $K_{<d}[X]$ l'ensemble des polynômes de $K[X]$ de degré strictement inférieur à d . C'est un espace vectoriel de dimension d . L'espace affine $X^d + K_{<d}[X]$ des polynômes unitaires de degré d sera noté $K_d[X]$.

Évaluation et interpolation

Après la somme et le produit, qui peuvent être traités par les méthodes déjà vues précédemment, deux opérations de base sur les polynômes sont l'évaluation et l'interpolation. L'évaluation correspond à la fonction $(P, x) \mapsto P(x)$, où P est un polynôme et $x \in K$. En différentiant, nous obtenons :

$$(P + dP)(x + dx) = P(x + dx) + dP(x + dx) = P(x) + P'(x)dx + dP(x). \quad (3.1)$$

Ici, P' est la dérivée de P . La différentielle de l'évaluation en P est alors l'application linéaire $(dP, dx) \mapsto P'(x)dx + dP(x)$.

Pour ce qui est de l'interpolation, nous fixons un entier positif d et voulons étudier l'application $f : K^{2d} \rightarrow K_{<d}[X]$ qui envoie $(x_1, y_1, \dots, x_d, y_d)$ sur le polynôme P de degré au plus $d - 1$ tel que $P(x_i) = y_i$ pour tout i . Un tel polynôme P existe et est unique dès que les x_i sont deux à deux distincts. Nous définissons alors la fonction f sur l'ouvert correspondant. En outre, si $(x_1, y_1, \dots, x_d, y_d)$ est un point de cet ouvert et P correspond au polynôme interpolé, l'équation (3.1) montre que $dy_i = P'(x_i)dx_i + dP(x_i)$ pour tout i . En conséquence, il est possible d'obtenir $dP(x_i)$ à partir de dx_i et dy_i et ainsi, d'obtenir finalement dP par interpolation. Il pourrait être intéressant alors de considérer le cadre de la Sous-Section 2.3.2 pour étudier plus précisément le comportement de la différentielle.

Division euclidienne

Soit A et B deux polynômes avec $B \neq 0$. Le problème de la division euclidienne de A par B consiste à trouver Q et R tels que $A = BQ + R$ et $\deg R < \deg B$. En différentiant cette égalité, nous trouvons :

$$dA - dB \cdot Q = B \cdot dQ + dR,$$

ce qui implique que dQ et dR sont respectivement obtenus comme quotient et reste de la division euclidienne de $dA - dB \cdot Q$ par B . Ceci nous donne la différentielle. Là encore, son étude rentrerait dans le cadre de la Sous-Section 2.3.2.

Nous remarquons aussi que la discussion précédente s'adapterait tout fait à l'étude de division sur des anneaux de séries convergentes (voir aussi [CL14]).

PGCD et coefficients de Bézout

Soit n et m deux entiers positifs avec $n \geq m$. Nous considérons la fonction $f : K_n[X] \times K_m[X] \rightarrow (K_{\leq n}[X])^3$ qui envoie le couple (A, B) sur le triplet (D, U, V) où D est le PGCD (unitaire) de A et B , et U et V sont les coefficients de Bézout, de degré minimaux et calculés par l'algorithme d'Euclide étendu. La non-annulation du résultant de A et B définit un ouvert de Zariski \mathcal{V}_0 où la fonction pgcd prend une valeur constante égale à 1. En-dehors de \mathcal{V}_0 , $\text{pgcd}(A, B)$ est un polynôme de degré strictement positif. Puisque \mathcal{V}_0 est dense, f n'est pas continu en-dehors de \mathcal{V}_0 . Par contre, f est différentiable, et même localement analytique sur \mathcal{V}_0 .

Bien sûr, sur \mathcal{V}_0 , la première composante de f , D , est constante et nous avons $dD = 0$. Pour calculer dU et dV , nous différencions simplement la relation de Bézout $AU + BV = 1$. Nous obtenons :

$$A \cdot dU + B \cdot dV = -(dA \cdot U + dB \cdot V)$$

d'où nous déduisons que dU (resp. dV) s'obtient comme le reste dans la division euclidienne de $U \cdot dX$ par B (resp. de $V \cdot dX$ par A) où $dX = -(dA \cdot U + dB \cdot V)$.

Afin de différencier f en-dehors de \mathcal{V}_0 , nous définissons le sous-ensemble \mathcal{V}_i de $K_n[X] \times K_m[X]$ comme le lieu où le pgcd a degré i . La théorie des sous-résultants montre que \mathcal{V}_i est localement fermé pour la topologie de Zariski. En particulier, \mathcal{V}_i est une K -variété au sens de la Section 2.5, et la restriction de f à \mathcal{V}_i est différentiable. Pour calculer cette différentielle, nous procédons comme précédemment en différenciant la relation $AU + BV = D$. Nous obtenons ainsi $A \cdot dU + B \cdot dV - dD = dX$ avec $dX = -(dA \cdot U + dB \cdot V)$. Dans cette relation, les deux premiers termes, $A \cdot dU$ et $B \cdot dV$ sont divisibles par D alors que dD a un degré strictement inférieur à i . Ainsi, si $dX = D \cdot dQ + dR$ est la division euclidienne de dX par D , nous devons avoir $\frac{A}{D} \cdot dU + \frac{B}{D} \cdot dV = dQ$ et $dD = -dR$. Ces relations, avec les bornes sur les degrés de U et V , impliquent que dU (resp. dV) est égal au reste dans la division euclidienne de $U \cdot dQ$ par $\frac{B}{D}$ (resp. de $V \cdot dQ$ par $\frac{A}{D}$).

Nous pouvons discuter de quelques enseignement du paragraphe précédent. Si nous devons calculer le pgcd de deux polynômes A et B connus à précision finie, nous devons d'abord déterminer son degré. Hélas, la fonction degré n'est pas continue (elle est seulement semi-continue supérieurement), et ainsi, ne peut être déterminée à partir de A et B , à moins que les approximations de (A, B) soit contenues dans \mathcal{V}_0 .

Une hypothèse supplémentaire doit être alors faite. La plus naturelle est de prendre $\text{pgcd}(A, B)$ du plus grand degré possible. La raison principale pour cette convention est que si les polynômes qui nous intéressent, $A, B \in K[X]$, ont un pgcd de degré i , alors il existe une précision à partir de laquelle ce degré maximal sera i . Une fois cette hypothèse faite, le calcul est réalisable et il est possible d'appliquer le Lemme 2.2.4 pour déterminer la précision sur le résultat. Remarquons qu'avec cette convention, le pgcd de A et A est A lui-même, et ce bien qu'il existe des couples de polynômes premiers entre eux dans tout voisinage de (A, A) .

Factorisation

Soit $P_0 \in K_d[X]$ un polynôme qui s'écrit $P_0 = A_0 B_0$, avec A_0 et B_0 unitaires et premiers entre eux. Le lemme de Hensel implique qu'il existe un voisinage \mathcal{U} de P_0 dans $K_d[X]$ tel que tout $P \in \mathcal{U}$ se factorise de manière unique en $P = AB$ avec A et B unitaires et proches de A_0 et B_0

respectivement. Ainsi, nous pouvons considérer l'application $f : P \mapsto (A, B)$ définie sur $\mathcal{U} \subset K_d[X]$. Nous souhaitons différentier f en P_0 . Pour cela, nous différencions l'égalité $P = AB$ en P_0 pour obtenir :

$$dP = A_0 \cdot dB + B_0 \cdot dA, \quad (3.2)$$

avec dP , dA et dB de degré au plus $\deg P - 1$, $\deg A - 1$ et $\deg B - 1$ respectivement. Nous pouvons remarquer que la matrice de cette application est la bien connue matrice de Sylvester. Si $A_0 U_0 + B_0 V_0 = 1$ est une relation de Bézout entre A_0 et B_0 , nous déduisons de (3.2) que dA (resp. dB) est le reste dans la division euclidienne de $V_0 \cdot dP$ par A_0 (resp. de $U_0 \cdot dP$ par B_0).

Recherche de racines

Un cas particulier important apparaît dans l'étude précédente lorsque A_0 est de degré 1, c'est-à-dire $A_0(X) = X - \alpha_0$ pour un certain $\alpha_0 \in K$. Ceci correspond bien sûr à la recherche de racine de A_0 . L'application $P \mapsto A$ consiste donc en suivre la racine simple α_0 . Ceci peut être fait grâce à l'étude précédente sur la factorisation. Néanmoins, nous pouvons donner une expression plus directe de la différentielle concernée en développant l'équation : $(P_0 + dP)(\alpha_0 + d\alpha) = 0$, ce qui donne $P'_0(\alpha_0)d\alpha + dP(\alpha_0) = 0$. Puisque α_0 est une racine simple, $P'_0(\alpha_0)$ ne s'annule pas et nous trouvons :

$$d\alpha = -\frac{dP(\alpha_0)}{P'_0(\alpha_0)}.$$

Nous pouvons maintenant étudier le cas d'une racine multiple. Soit P_0 un polynôme unitaire de degré d et $\alpha_0 \in K$ une racine de P_0 de multiplicité $m > 1$. Du fait de sa multiplicité, il n'est plus possible de suivre directement la racine α_0 sur un voisinage de P_0 . Cependant, nous pouvons tout de même nous restreindre aux polynômes qui ont une racine de multiplicité m au voisinage de α_0 . Plus précisément, soit \mathcal{V}_m l'ensemble des polynômes unitaires de degré d ayant une racine de multiplicité au moins m . Il s'agit d'un fermé de $K_d[X]$ pour la topologie de Zariski, et il contient, par hypothèse, le polynôme P_0 . En outre, les composantes irréductibles de \mathcal{V}_m qui rencontrent P_0 sont en bijection avec l'ensemble des racines de P_0 de multiplicité $\geq m$. En particulier, α_0 correspond à l'une de ces composantes irréductibles. Notons-là \mathcal{V}_{m,α_0} . La variété algébrique \mathcal{V}_{m,α_0} est *a fortiori* une K -variété. En outre, il existe une fonction différentiable f qui est définie sur un voisinage de P_0 dans \mathcal{V}_{m,α_0} et suit la racine α_0 , i.e. $f(P_0) = \alpha_0$ et pour tout P tel que $f(P)$ est défini, $\alpha = f(P)$ est une racine de multiplicité (au moins) m de P . L'existence de f suit du lemme de Hensel appliqué à la factorisation $P_0(X) = (X - \alpha_0)^m B_0(X)$ où $B_0(X)$ est un polynôme qui est premier avec $X - \alpha_0$. Nous pouvons donc finalement calculer la différentielle de f en P_0 (selon \mathcal{V}_{m,α_0}) par différentiation de l'égalité $P^{(m-1)}(\alpha) = 0$ où $P^{(i)}$ est la dérivée i -ème de P . Nous trouvons :

$$d\alpha = -\frac{dP^{(m-1)}(\alpha_0)}{P_0^{(m)}(\alpha_0)}. \quad (3.3)$$

Le calcul précédent a des conséquences intéressantes pour ce qui est de la précision p -adique. Par exemple, si l'on considère le polynôme unitaire $P(X) = X^2 + O(p^{2N})X + O(p^{2N})$ (où N est un entier grand) et que l'on souhaite calculer sur ordinateur une de ses racines, quelle réponse peut-on attendre ? Nous remarquons que, si α est un nombre p -adique divisible par p^N , alors parmi $X^2 + O(p^{2N})X + O(p^{2N})$, on trouve $X^2 - \alpha^2$ dont les racines sont $\pm\alpha$. Réciproquement, nous pouvons prouver que les deux racines de P sont nécessairement divisibles par p^N . Ainsi, une bonne réponse est certainement $O(p^N)$. Néanmoins, si nous avons l'information supplémentaire que P a une racine double, disons α , alors, nous pouvons écrire $P(X) = (X - \alpha)^2$ et en identifiant α avec le coefficient en X , nous obtenons $\alpha = O(p^{2N})$ si $p > 2$ et $\alpha = O(p^{2N-1})$ si $p = 2$. Il s'agit exactement du résultat que nous obtenons en appliquant le Lemme 2.2.4 et en utilisant (3.3) pour simplifier l'écriture de la différentielle.

Ce phénomène est vrai en général : si P est un polynôme sur \mathbb{Q}_p connu à précision finie $O(p^N)$, une racine α de P ayant possiblement une racine de multiplicité m (i.e. $P(\alpha), P'(\alpha), \dots, P^{(m-1)}(\alpha)$ sont nuls à la précision donnée) peut être calculée à précision environ $O(p^{N/m})$. Cependant, si nous avons l'information supplémentaire que la multiplicité est exactement m , nous pouvons calculer cette racine à précision $O(p^{N-c})$ avec c une constante dépendant de P et de α mais pas de N .

Division multivariée

Nous étudions l'anneau $K[\mathbf{X}] = K[X_1, \dots, X_n]$ des polynômes en n variables sur K et choisissons un ordre monomial sur $K[\mathbf{X}]$. Nous renvoyons à la Section 6.1 pour plus de détails sur de telles notions.

Il existe alors une notion de division sur $K[\mathbf{X}]$: si f, f_1, \dots, f_s sont des polynômes dans $K[\mathbf{X}]$, alors il est possible d'écrire f comme

$$f = q_1 f_1 + \dots + q_s f_s + r$$

où $q_1, \dots, q_s, r \in K[\mathbf{X}]$ et aucun monôme de r n'est divisible par le monôme de tête d'un f_i ($1 \leq i \leq s$). Nous disposons d'un algorithme pour calculer cette division, qui est décrit dans l'Algorithme 6.1.9. Par ailleurs, si l'on suppose que (f_1, \dots, f_s) est une base de Gröbner¹ de l'idéal engendré par (f_1, \dots, f_s) , le polynôme r est uniquement déterminé et appelé le *reste* dans la division de f par la famille (f_1, \dots, f_s) . L'application $(f, f_1, \dots, f_s) \mapsto r$ est alors bien définie et nous pouvons calculer sa différentielle : nous trouvons que dr est le reste dans la division de $f - (q_1 \cdot df_1 + \dots + q_s \cdot df_s)$ par (f_1, \dots, f_s) .

3.2. Matrices

Dans cette section, nous nous intéressons à diverses opérations sur les matrices à coefficients dans K : multiplication, calcul du déterminant, du polynôme caractéristique et calcul de la décomposition LU. Nous y étudierons plus précisément l'intérêt qu'apporte l'utilisation de réseaux comme donnée de précision comparé à une approche naïve.

3.2.1. Multiplication

Pour commencer, nous étudions le comportement de la précision lorsqu'on effectue des multiplications matricielles. Soit r, s et t trois entiers strictement positifs. On suppose que l'on souhaite multiplier une matrice $A \in M_{r,s}(K)$ par une matrice $B \in M_{s,t}(K)$. Cette opération est, bien sûr, donnée par une fonction polynomiale entière :

$$\begin{aligned} \mathcal{P}_{r,s,t} : M_{r,s}(K) \times M_{s,t}(K) &\rightarrow M_{r,t}(K) \\ (A, B) &\mapsto AB. \end{aligned}$$

D'après la Proposition 2.2.4, le comportement de la précision lorsqu'on calcule AB est donné par $\mathcal{P}'_{r,s,t}(A, B)$, l'application linéaire qui envoie le couple (dA, dB) sur $A \cdot dB + dA \cdot B$.

Pour simplifier, supposons que les coefficients de A et B sont tous dans \mathcal{O}_K et connus à la même précision $O(\pi^N)$. Afin d'appliquer les Propositions 2.2.4 et 2.3.14, nous avons besoin de calculer l'image du réseau standard $\mathcal{L}_0 = M_{r,s}(\mathcal{O}_K) \times M_{s,t}(\mathcal{O}_K)$ par $\mathcal{P}'_{r,s,t}(A, B)$. Il est bien sûr contenu dans $M_{r,t}(\mathcal{O}_K)$. Ceci reflète le fait évident que chaque entrée du produit AB est connue à précision au moins $O(\pi^N)$. Néanmoins, il peut arriver que l'inclusion précédente soit stricte, ce qui signifie que l'on *gagne* de la précision dans ces cas.

Posons $a_i = \sigma_i(A)$ et $b_i = \sigma_i(B)$ (valuation du i -ème facteur invariant), et définissons $M_{r,t}((a_i), (b_j))$ comme le sous-réseau de $M_{r,t}(\mathcal{O}_K)$ formé de matrices $M = (M_{i,j})$ telles que $\text{val}(M_{i,j}) \geq \min(a_i, b_j)$ pour tout (i, j) . Écrivons $a^{(m)}$ (resp. $b^{(m)}$) pour le nombre de a_i (resp. b_i) qui sont au moins m .

Proposition 3.2.1. *Avec les notations précédentes, nous avons :*

$$\begin{aligned} \mathcal{P}'_{r,s,t}(A, B)(\mathcal{L}_0) &= U_A \cdot M_{r,t}((a_i), (b_j)) \cdot V_B \\ \text{et } \text{longueur} \left(\frac{M_{r,t}(\mathcal{O}_K)}{\mathcal{P}'_{r,s,t}(A, B)(\mathcal{L}_0)} \right) &= \sum_{m=1}^{\infty} a^{(m)} b^{(m)}. \end{aligned}$$

1. Un choix canonique pour r existe même sans cette hypothèse, donné par exemple par le résultat de l'Algorithme 6.1.9, mais il n'est pas clair que le calcul que nous effectuons de la différentielle s'étende à ce cadre.

Démonstration. Écrivons $A \cdot dB + dA \cdot B = U_A \cdot M \cdot V_B$ avec

$$M = \Delta_A \cdot V_A \cdot dA \cdot V_B^{-1} + U_A^{-1} \cdot dB \cdot U_B \cdot \Delta_B.$$

Lorsque dA varie dans $M_{r,s}(\mathcal{O}_K)$ il en va de même pour $V_A \cdot dA \cdot V_B^{-1}$ et ainsi, le premier terme dans l'écriture ci-dessus de M varie dans le sous-espace de $M_{a,s}(\mathcal{O}_K)$ formé des matrices dont la i -ème ligne a valuation au moins a_i . Avec un argument similaire pour le second terme de la somme nous en déduisons le résultat de la première partie de la proposition. La seconde partie est laissée au lecteur. \square

Du point de vue de la précision, la deuxième formule de la Proposition 3.2.1 veut dire que le calcul de AB augmente la précision absolue de $\sum_{m=1}^{\infty} a^{(m)}b^{(m)}$ chiffres² dès que $N > \min(a_r, b_t)$ (cf. Proposition 2.3.14). Cependant, beaucoup de ces chiffres sont diffusés au sens de la Définition 2.1.3. Nous pouvons faire un changement de bases afin de visualiser cette augmentation de précision en coordonnées : écrivons $AB = U_A \cdot P \cdot V_B$ avec $P = \Delta_A \cdot V_A \cdot U_B \cdot \Delta_B$. Si l'on suit la précision de manière classique, le coefficient d'indice (i, j) de P est connu à précision $O(\pi^{N+\min(a_i, b_j)})$. Les multiplications par U_A et V_B diffusent alors la précision sur les coefficients de AB .

Nous allons maintenant considérer l'impact d'un suivi précis de cette précision diffuse. Bien que le bénéfice n'est pas substantiel pour un unique produit de matrices, il s'accumule si l'on fait un nombre important de multiplication de matrices. Nous illustrons ce phénomène avec l'exemple simple suivant :

Algorithme 3.2.2 : Exemple de produits

Entrée : une liste (M_1, \dots, M_n) de matrices carrées de taille d , avec coefficients connus à précision $O(\pi^n)$.

Sortie : Le coefficient $(1, 1)$ de $\prod_{i=1}^n M_i$.

début

P reçoit la matrice identité de taille d ;
pour $j = 1, \dots, n$ **faire**
Faire $P = PM_j$;
Retourner le coefficient $(1, 1)$ de P ;

Le tableau en Figure 3.1 en page 74 compare le nombre de chiffres significatifs en précision *relative* que nous perdons en sortie de l'Algorithme 3.2.2 lorsque l'on utilise d'un côté un suivi de précision classique, coordonnée par coordonnée, et de l'autre côté une méthode fondée sur les réseaux. Dans le premier cas, nous utilisons l'implémentation classique du produit et des p -adiques en Sage. Dans le second cas, à la matrice P de l'Algorithme 3.2.2 est attaché un réseau dP dans $M_d(K)$, et lorsque nous mettons à jour P avec $P \cdot M_i$, nous faisons de même pour dP en posant $dP \leftarrow dP * M_i + M_d(p^n * \mathcal{O}_K) * P$. Ceci est évidemment plus coûteux en temps de calcul par un facteur constant, mais nous allons voir que le gain en terme de précision peut être loin d'être négligeable.

Nous observons que, dans le premier cas, le nombre de chiffres perdus semble croître linéairement avec le nombre de multiplications effectuées (ici notée n), tandis que dans le second cas, cette croissance semble n'être que logarithmique. Nous apprécierions de connaître une formulation plus précise et une preuve de cette heuristique.

Remarque 3.2.3. S'intéresser à un produit de matrices aléatoires n'est pas anodin. Il s'agit de l'objet d'étude central de la théorie des marches aléatoires sur des espaces homogènes, qui peuvent en particulier être de nature p -adique, voire S -adique ($S \subset \infty \cup \{p, p \text{ premier}\}$ fini). Nous renvoyons à [BQ12] de Benoist et Quint pour une introduction à ce domaine. Être capable d'estimer numériquement de manière précise et efficace des exposants de Lyapunov grâce à des variantes de l'Algorithme 3.2.2 et l'usage de réseaux pour suivre la précision peut alors y être particulièrement utile.

2. Nous notons néanmoins que la valuation des entrées de AB peuvent augmenter, ce qui veut dire que nous pouvons perdre des chiffres significatifs, si l'on raisonne en terme de précision relative.

3. Applications du lemme, calcul de différentielles

| d | n | Perte de précision moyenne | |
|-----|------|----------------------------|-------------------|
| | | Suivi par coordonnées | Suivi par réseaux |
| 2 | 10 | 2, 8 | 2, 4 |
| 2 | 100 | 16, 7 | 5, 0 |
| 2 | 1000 | 157, 8 | 7, 9 |
| 3 | 10 | 2, 2 | 1, 9 |
| 3 | 100 | 12, 8 | 4, 0 |
| 3 | 1000 | 122, 5 | 7, 0 |

Résultats pour un échantillon de 1000 entrées aléatoires dans $M_{d,d}(\mathbb{Z}_2)^n$

FIGURE 3.1. – Perte de précision moyenne dans l’Algorithme 3.2.2

3.2.2. Déterminant

Le calcul de la différentielle de l’application déterminant $\det : M_{n,n}(K) \rightarrow K$ est classique : en un point M , il s’agit de l’application linéaire

$$\det'(M) : dM \mapsto \text{Tr}(\text{Com}(M) \cdot dM),$$

où $\text{Com}(M)$ est la comatrice de M , qui correspond à $\det(M)M^{-1}$ lorsque M est inversible. Si M est de rang n ou $n-1$, alors $\text{Com}(M)$ est de rang n ou 1 respectivement, et $\det'(M)$ est surjective. Si $\text{rang}(M) \leq n-2$, alors $\text{Com}(M) = 0$ et $\det'(M)$ est nulle, donc *a fortiori* non surjective et nous ne pourrions pas appliquer le Lemme 2.2.4. Ainsi, nous supposons que $\text{rang}(M) \geq n-1$ dans cette Sous-Section.

Comme pour le cas de la multiplication de matrices, nous calculons d’abord l’image du réseau standard $\mathcal{L}_0 = M_{n,n}(\mathcal{O}_K)$ par $\det'(M)$.

Proposition 3.2.4. *Posons $v = \sigma_1(M) + \dots + \sigma_{n-1}(M)$, nous avons $\det'(M)(\mathcal{L}_0) = \pi^v \mathcal{O}_K$.*

Démonstration. Avec la description explicite de $\det'(M)$, nous voyons directement qu’il suffit de montrer que le coefficient de $\text{Com}(M)$ de plus petite valuation a valuation v , ou, de manière équivalente, que l’idéal de \mathcal{O}_K engendré par les mineurs de M de taille $(n-1)$ est $\pi^v \mathcal{O}_K$. Ceci est dû au fait suivant : cet idéal est inchangé lorsqu’on multiplie à gauche ou à droite par une matrice de $GL_n(\mathcal{O}_K)$. Ainsi, nous pouvons supposer que $M = \Delta_M$, et le résultat est alors clair. \square

Du point de vue de la précision, la Proposition 3.2.4 implique que si M est connue à précision uniforme en entrée $O(\pi^N)$ avec $N > v$, alors $\det(M)$ est connu à précision $O(\pi^{N+v})$. Ainsi, nous gagnons v chiffres de précision absolue, ou de manière équivalente, perdons $\sigma_n(M)$ chiffres de précision relative (ce qui peut être $= \infty$ lorsque $\det M = 0 \dots$). En outre, il est possible de calculer $\det(M)$ à cette précision optimale en calculant d’abord une réalisation d’une forme normale de Smith approchée de M par l’Algorithme 1.3.5, $M = P\Delta_M Q$ avec les coefficients de P et Q connus à précision $O(\pi^N)$. Comme de plus, $\det P = \det Q = \pm 1$, et que Δ_M est sous forme diagonale à $O(\pi^N)$ près, nous pouvons lire le résultat directement sur Δ_M .

Exemple 3.2.5. Nous adaptons l’exemple 1.3.13 en étudiant

$$M = \begin{bmatrix} 2 + O(2^6) & 4 + O(2^6) & 4 + O(2^6) \\ -6 + O(2^6) & 6 + O(2^6) & 12 + O(2^6) \\ 10 + O(2^6) & -4 + O(2^6) & -16 + O(2^6) \end{bmatrix}.$$

En développant directement, on obtient $\det M = -144 + O(2^6) = 3 * 2^4 + O(2^6)$. Cependant, nous obtenons par l’Algorithme 1.3.5

$$\Delta_M = \begin{bmatrix} 2 + O(2^6) & O(2^6) & O(2^6) \\ O(2^6) & 2 + O(2^6) & O(2^6) \\ O(2^6) & O(2^6) & 4 + O(2^6) \end{bmatrix},$$

avec $PMQ = \Delta_M$ et $\det P = -\frac{1}{9}$ et $\det Q = 1$, ce qui donne $\det M = -9 * 2^4 + O(2^8) = 3 * 2^4 + 2^6 + O(2^8)$.

3.2.3. Polynôme caractéristique

Écrivons $\text{char} : M_{n,n}(K) \rightarrow K[X]$ pour noter le polynôme caractéristique, et $K_{<n}[X] \subset K[X]$ pour le sous-espace constitué des polynômes de degré strictement inférieur à n . Alors la différentielle de char au point M est

$$\text{char}'(M) : dM \mapsto \text{Tr}(\text{Com}(X - M) \cdot dM).$$

L'image de $M_{n,n}(K)$ par $\text{char}'(M)$ est le K -espace vectoriel engendré par les coefficients de $\text{Com}(X - M)$, qui est clairement inclus dans $K_{<n}[X]$. Plus précisément, l'image est égale à $K_{<n}[X]$ dès que M n'a pas deux blocs de Jordan pour une même valeur propre.

Nous rappelons que le *polygone de Newton* $\text{NP}(f)$ d'un polynôme $f(X) = \sum_i a_i X^i$ est l'enveloppe convexe inférieure des points $(i, \text{val}(a_i))$ et le *polygone de Newton* $\text{NP}(M)$ d'une matrice M est $\text{NP}(\text{char}(M))$. Le *polygone de Hodge* de M est l'enveloppe convexe inférieure des points $(i, \sum_{j=1}^{n-i} \sigma_j(M))$. Pour une matrice M , le polygone de Newton est au-dessus du polygone de Hodge, voir [Ked10] Thm. 4.3.11.

Nous avons déjà vu que de tels polynômes apparaissent naturellement lorsque l'on s'intéresse à la précision, voir par exemple 2.3.1. Un tel polygone P engendre un réseau \mathcal{L}_P dans $K_{<n}[X]$ formé des polynômes dont le polygone de Newton se trouve au-dessus de celui de P . Ce réseau est engendré par les monômes $a_i X^i$, avec $\text{val}(a_i)$ la partie entière supérieure de la hauteur de P à i . Il est possible d'utiliser ces polygones pour faire un certain type de suivi de la précision sur les coordonnées, donnant une borne sur le réseau de précision image de char' :

Définition 3.2.6. Le *polygone de précision* $\text{PP}(M)$ de M est l'enveloppe convexe inférieure des polygones de Newton des coefficients de $\text{Com}(X - M)$.

Il est clair, vu la définition, que $\mathcal{L}_{\text{PP}(M)} \subset \text{char}'(M)(\mathcal{L}_0)$ où \mathcal{L}_0 est le réseau standard $M_{n,n}(\mathcal{O}_K)$. Plus précisément $\text{PP}(M)$ est le plus petit polygone P pour lequel l'inclusion $\mathcal{L}_P \subset \text{char}'(M)(\mathcal{L}_0)$ est satisfaite. Par le Lemme 2.2.4, le polygone de précision détermine la perte de précision minimale que l'on peut déterminer en regardant des polygones.

Il se trouve que le polygone de précision est lié aux polygones de Hodge et de Newton. Si un polygone P a pour sommets (x_i, y_i) , soit $T_n(P)$ le translaté par n du polygone, avec pour sommets $(x_i - n, y_i)$.

Proposition 3.2.7. Le polygone de précision $\text{PP}(M)$ se trouve entre $T_1(\text{HP}(M))$ et $\text{NP}(M)$.

De plus, $\text{PP}(M)$ et $T_1(\text{HP}(M))$ coïncident en 0 et $n-1$.

Démonstration. Les coefficients de $\text{char}(M)$ peuvent être exprimés comme trace de puissances extérieures : le coefficient de X^i est $\text{Tr}(\Lambda^i(M))$, qui est $\text{Tr}(\Lambda^i(U_M) \Lambda^i(\Delta_M) \Lambda^i(V_M))$. En calculant $\Lambda^i(\Delta_M)$, nous obtenons la première partie de la Proposition. Pour $i = 1$, nous trouvons de plus que $\text{PP}(M)$ s'annule pour l'abscisse $n-1$. Par définition, c'est aussi le cas de $T_1(\text{HP}(M))$. Le fait que $\text{PP}(M)$ et $T_1(\text{HP}(M))$ coïncident pour l'abscisse 0 est conséquence de la Proposition 3.2.4.

Il reste à comparer avec le polygone de Newton. Posons $f = \text{char}(M)$, $m_{i,j}$ le coefficient d'indice (i, j) de M , $f_{i,j}$ le coefficient d'indice (i, j) $\text{Com}(X - M)$ et $\mu_{i,j} = \text{val}(m_{i,j})$. Nous notons $f[k]$ pour la valuation du coefficient de X^k dans f , et posons $f[-1] = +\infty$. L'équation $(X - M) \cdot \text{Com}(X - M) = f \cdot I$ implique, pour tout i et k ,

$$\begin{aligned} f[k] &\geq \inf(f_{i,i}[k-1], \mu_{i,0} + f_{0,i}[k], \dots, \mu_{i,n} + f_{n,i}[k]) \\ &\geq \inf(f_{i,i}[k-1], f_{j,i}[k]), \end{aligned}$$

avec le minimum pris sur j . En prenant l'enveloppe convexe inférieure et en remarquant que $\text{PP}(M)$ est décroissant, ce qui suit de la comparaison avec le polygone de Hodge, nous obtenons le résultat. \square

Remarque 3.2.8. Diverses expériences numériques nous suggèrent le fait, plus fort, que $\text{PP}(M)$ est borné supérieurement par $T_1(\text{NP}(M))$.

Pour beaucoup de matrices, $\text{PP}(M) = T_1(\text{HP}(M))$. Pour 500 tirages de matrices aux coefficients pris aléatoirement dans \mathbb{Z}_2 , le polygone de précision 2-adique est égal au polygone de Hodge dans

3. Applications du lemme, calcul de différentielles

99.5% des cas en dimension 4, et dans 99.1% en dimension 8. Sur \mathbb{Z}_3 , ce pourcentage monte à 99.98%, sans dépendance claire sur la dimension. Empiriquement, $\text{PP}(M)$ semble avoir le plus de chance de différer de $T_1(\text{HP}(M))$ en 1, correspondant à la précision sur le terme linéaire du polynôme caractéristique.

Bien sûr, le réseau de précision $\mathcal{E} = \text{char}'(M)(\mathcal{L}_0)$ peut contenir de la précision diffuse qui ne serait pas contenue dans $\text{PP}(M)$. Pour 500 tirages, de la précision diffuse apparaît dans 11% des cas en dimension 3, jusqu'à 15% des cas en dimension 8. Ce pourcentage augmente lorsque $\text{val}(\det(M))$ décroît, atteignant 34% en dimension 9 pour les matrices ayant la contrainte d'avoir un déterminant de valuation 2-adique 12. En outre, il est possible de construire des exemples avec arbitrairement autant de précision diffuse que souhaité. Supposons que $\sigma_i(M)$ soit grand. Alors la Proposition 3.2.7 implique que \mathcal{E} est contenu dans $\mathcal{O}_{K, < n}[X]$ avec indice au moins $\sum_{i=1}^{n-1} \sigma_k(M)$. Le réseau de précision de $1 + M$ est obtenue à partir de \mathcal{E} avec la transformation $X \mapsto 1 + X$, mais nous avons $\text{PP}(1 + M)$ qui est aplati, de hauteur 0.

En conclusion, pour des matrices choisies aléatoirement, approcher \mathcal{E} en utilisant le polygone de Hodge amène une perte de précision par rapport au réseau de précision assez faible, mais non nulle. Il est possible que cette perte puisse s'accumuler lorsqu'on effectue des opérations sur les polynômes. Par ailleurs, si les $\sigma_i(M)$ sont connus pour être assez grand ou si M est le translaté d'une telle matrice, utiliser le réseau de précision peut être utile pour estimer la perte de précision.

3.2.4. Décomposition LU

Dans cette Sous-Section, nous noterons par $\|\cdot\|$ une norme sur $M_n(K)$ subordonnée à une norme $\|\cdot\|$ sur K^n . Pour un réel C , nous notons $B(C)$ la boule fermée de $M_n(K)$ centrée en 0 et de rayon C . Nous considérerons aussi les sous-ensembles suivants de $M_n(K)$:

- O_n est l'ouvert des matrices dont les mineurs principaux sont non nuls (nous rappelons que cette condition implique existence et unicité de la décomposition LU) ;
- U_n est le sous-espace vectoriel de $M_n(K)$ formé des matrices triangulaires supérieures ;
- L_n^0 (resp. L_n^u) le sous-espace affine de $M_n(K)$ formé des matrices triangulaires inférieures nilpotentes (resp. unipotentes).

Analyse et précision

Nous choisissons de normaliser la décomposition LU en demandant que L soit unipotente et nous notons $D : O_n \rightarrow L_n^u \times U_n$ la fonction correspondant à cette décomposition sur l'ouvert O_n . Pour $M \in O_n$ avec $D(M) = (L, U)$, l'application linéaire $D'(M)$ est donnée par :

$$dM \mapsto (L \cdot \text{low}(dX), \text{up}(dX) \cdot U) \text{ avec } dX = L^{-1} \cdot dM \cdot U^{-1}$$

avec low (resp. up) qui est la projection canonique de $M_n(K)$ sur L_n^0 (resp. U_n). Il est facile de montrer que $D'(M)$ est bijective, d'inverse donnée par $(A, B) \mapsto AU + LB$.

Nous souhaitons maintenant appliquer la Proposition 2.3.16 afin d'obtenir un résultat complètement effectif sur la précision. *Nous supposons ici que K est de caractéristique 0.* Soit $M_0 \in O_n$ et posons $D(M_0) = (L_0, U_0)$. Nous considérons la fonction translatée f envoyant M sur $D(M_0 + M) - D(M_0)$. Nous avons alors $f(0) = 0$ et $f'(M) = D'(M_0 + M)$. En utilisant une description explicite de l'inverse de $D'(M_0 + M)$, nous trouvons $B(1) \subset f'(0) \cdot B(C)$ pour $C = \max(\|U_0\|, \|L_0\|)$. En outre, f satisfait l'équation différentielle $f' = g \circ f$ avec g donné par :

$$\begin{aligned} g(A, B)(X) &= ((L_0 + A) \cdot \text{low}(Y), \text{up}(Y) \cdot (U_0 + B)) \\ &\text{avec } Y = (L_0 + A)^{-1} \cdot X \cdot (U_0 + B)^{-1}. \end{aligned} \quad (3.4)$$

Soit $\kappa(S) = \|S\| \cdot \|S^{-1}\|$ le conditionnement de la matrice S . Remarquons que $\|S + T\| = \|S\|$ si $\|T\| < \|S\|$ et $\|(S + T)^{-1}\| = \|S^{-1}\|$ si $\|T\| < \|S^{-1}\|^{-1}$. Nous déduisons de (3.4) que

$$\|g(A, B)\| \leq \max(\kappa(L_0)\|U_0^{-1}\|, \kappa(U_0)\|L_0^{-1}\|)$$

dès que $\|A\| < \|L_0^{-1}\|^{-1}$ et $\|B\| < \|U_0^{-1}\|^{-1}$. En combinant ceci avec les Propositions 2.3.16 et 2.3.8,

nous trouvons finalement que Eq. (2.3) est satisfaite dès que

$$\frac{\rho}{r} > \|p\|^{-\frac{2p}{p-1}} \cdot \max(\|L_0\|, \|U_0\|) \cdot \max(\|L_0^{-1}\|, \|U_0^{-1}\|) \cdot \max(\kappa(L_0)\|U_0^{-1}\|, \kappa(U_0)\|L_0^{-1}\|)^2.$$

Expériences numériques

Soit $B_n = (E_{i,j})_{1 \leq i,j \leq n}$ la base canonique de $M_n(K)$. Elle peut naturellement être aussi vue comme une base de $L_n^0 \times U_n$. Pour un $M \in O_n$ donné, nous abuserons des notations et écrirons $D'(M)$ pour la matrice de l'application linéaire dans cette base. Une remarque intéressante est que $D'(M)$ est triangulaire inférieure dans cette base³. En projetant $D'(M)$ sur chaque coordonnée, nous obtenons que la meilleure perte de précision que l'on peut obtenir en suivant la perte de précision coordonnée par coordonnée dans le calcul de D est donnée par $\sum_u (\max_v (\text{val}(D'(M)_{u,v})))$. Nous comparons cette quantité à $\text{val}(\det(D'(M)))$, qui est la perte de précision que l'on obtient en suivant la perte de précision par les réseaux. Le nombre de chiffres de précision diffusé de $D'(M)(M_{n,n}(\mathcal{O}_K))$ est alors la différence entre ces deux nombres. Le tableau en Figure 3.2 page 77 consigne moyenne et écart-type obtenus sur un échantillon de 2000 matrices aléatoires dans $M_{d,d}(\mathbb{Z}_2)$.

| taille des matrices | Perte de précision dans la décomposition LU | | | |
|---------------------|---|------------|-------------------|------------|
| | suivi par coordonnées | | suivi par réseaux | |
| | moyenne | écart-type | moyenne | écart-type |
| 2 | 3,0 | 5 | 1,5 | 1,4 |
| 3 | 9,4 | 11 | 2,3 | 2,3 |
| 4 | 20 | 20 | 3,8 | 3,1 |

Résultat pour un échantillon de 2000 matrices

FIGURE 3.2. – Perte de précision dans la décomposition LU

3.3. Espaces vectoriels

Les espaces vectoriels sont généralement représentés comme des sous-espaces de K^n pour un certain n , et ainsi, apparaissent naturellement comme des points sur des grassmanniennes. Ainsi, il est possible d'utiliser le cadre développé dans la Section 2.5 pour étudier la précision dans ce contexte.

3.3.1. Géométrie des grassmanniennes

Étant donné E , un espace vectoriel de dimension fini sur K , et d , un entier dans $\llbracket 0, \dim E \rrbracket$, nous notons $\text{Grass}(E, d)$ pour la grassmannienne de dimension d de E . Il est bien connu que $\text{Grass}(E, d)$ a naturellement une structure de K -variété. Le but de cette Sous-Section est de rappeler quelques propriétés de sa géométrie. Dans ce qui suit, nous posons $n = \dim E$ et choisissons une base préférentielle pour E .

Description et espace tangent

Soit V un sous-espace vectoriel de E de dimension d . La grassmannienne $\text{Grass}(E, d)$ peut être vue comme le quotient de l'ensemble des applications linéaires injectives $f : V \hookrightarrow E$ modulo

3. Il est possible de s'en convaincre en regardant la manière dont fonctionne l'algorithme classique de calcul de la décomposition LU par échelonnement en lignes sans choix du pivot : il apparaît clairement que certains coefficients de L ou U sont indépendants de certains coefficients de M . Par exemple, ceux de la première ligne de L sont indépendants de la dernière ligne de M .

3. Applications du lemme, calcul de différentielles

l'action (par pré-composition) de $\text{GL}(V)$: l'application f représente son image $f(V)$. Il suit de cette description que l'espace tangent de $\text{Grass}(E, d)$ est canoniquement isomorphe à

$$\text{Hom}(V, E)/\text{End}(V) \simeq \text{Hom}(V, E/V).$$

Cartes

Soit V et V' deux sous-espaces supplémentaires de E (i.e. $V \oplus V' = E$). Nous supposons que V a dimension d et noterons par π la projection $E \rightarrow V$ correspondant à la décomposition précédente. Soit $\mathcal{U}_{V,V'}$ l'ensemble de toutes les injections $f : V \hookrightarrow E$ telles que $\pi \circ f = \text{id}_V$. Clairement, il s'agit d'un sous-espace affine de $\text{Hom}(V, E)$ dont la partie linéaire est isomorphe à $\text{Hom}(V, V')$. En outre, nous pouvons l'inclure dans $\text{Grass}(E, d)$ en envoyant $f \in \mathcal{U}_{V,V'}$ sur son image. De cette manière, $\mathcal{U}_{V,V'}$ apparaît comme un sous-espace ouvert de $\text{Grass}(E, d)$ formé exactement des sous-espaces vectoriels W de E de dimension d tels que $W \cap V' = 0$. En conséquence, l'espace tangent en un tel W devient isomorphe à $\text{Hom}(V, V')$. L'identification $\text{Hom}(V, V') \rightarrow \text{Hom}(W, E/W)$ est donnée par $du \mapsto (du \circ f^{-1}) \bmod W$ où $f : V \xrightarrow{\sim} W$ est donnée par l'application linéaire définissant W .

Lorsque le couple (V, V') varie, les sous-espaces ouverts $\mathcal{U}_{V,V'}$ recouvrent la grassmannienne et définissent un atlas. Lorsqu'on représente les espaces vectoriels sur ordinateurs, il est courant de se restreindre au sous-atlas constitué des cartes de la forme (V_I, V_{I^c}) où I parcourt la famille des sous-espaces de $\{1, \dots, n\}$ de cardinal d et V_I est le sous-espace engendré par les e_i avec $i \in I$. Un sous-espace $W \subset E$ appartient alors à au moins un $\mathcal{U}_{V_I, V_{I^c}}$ et, pour une famille donnée de générateurs de W , nous pouvons déterminer un tel I en même temps qu'une inclusion $f : V_I \hookrightarrow E$ en faisant un échelonnement en lignes de la matrice des générateurs de W .

Une variante

D'une autre manière, il est aussi possible de décrire $\text{Grass}(E, d)$ comme l'ensemble des applications linéaires surjectives $f : E \rightarrow E/V$ modulo l'action (par post-composition) de $\text{GL}(E/V)$. Cette identification permet d'écrire l'espace tangent en un point V comme le quotient $\text{Hom}(E, E/V)/\text{End}(E/V) \simeq \text{Hom}(V, E/V)$. Étant donnée une décomposition en somme directe $E = V \oplus V'$ comme plus haut, nous posons $\mathcal{U}_{V,V'}^*$ pour l'ensemble des applications linéaires $f : E \rightarrow V'$ dont la restriction à V' est l'identité. Il s'agit d'un sous-espace affine de $\text{Hom}(E, V')$ de partie linéaire $\text{Hom}(V, V')$ et qui peut être identifié à un ouvert de $\text{Grass}(E, d)$ via l'application $f \mapsto \ker f$.

Il est aisé de voir que $\mathcal{U}_{V,V'}$ et $\mathcal{U}_{V,V'}^*$ définissent le même ouvert de $\text{Grass}(E, d)$. En effet, étant donné $f \in \mathcal{U}_{V,V'}$, nous pouvons écrire $f = \text{id}_V + h$ avec $h \in \text{Hom}(V, V')$ et définir le morphisme $g = \text{id}_E - h \circ \pi \in \mathcal{U}_{V,V'}^*$. L'application $f \mapsto g$ définit alors une bijection $\mathcal{U}_{V,V'} \rightarrow \mathcal{U}_{V,V'}^*$ qui commute avec les inclusions dans la grassmannienne.

Dualité

Si E est un espace vectoriel de dimension finie sur K , nous utiliserons la notation E^* pour son dual (i.e. $E^* = \text{Hom}(E, K)$). Si nous avons aussi un sous-espace $V \subset E$, nous noterons par V^\perp le sous-espace de E^* formé des applications linéaires qui s'annulent sur V . Nous rappelons que le dual de V^\perp (resp. E^*/V^\perp) est canoniquement isomorphe à E/V (resp. V). Pour tout d , l'application $V \mapsto V^\perp$ définit un morphisme continu $\psi_E : \text{Grass}(E, d) \rightarrow \text{Grass}(E^*, n - d)$. L'action de ψ_E sur les espaces tangents peut se décrire aisément, et en particulier, la différentielle de ψ_E en V n'est rien d'autre que l'identification canonique entre $\text{Hom}(V, E/V)$ et $\text{Hom}(V^\perp, E^*/V^\perp)$ induite par la transposition. En outre, nous remarquons que ψ_E respecte les cartes que nous avons définies plus haut au sens suivant : il envoie bijectivement $\mathcal{U}_{V,V'}$ sur $\mathcal{U}_{V^\perp, (V')^\perp}^* \simeq \mathcal{U}_{V^\perp, (V')^\perp}$.

3.3.2. Calcul différentiel

Dans cette Sous-Section, nous allons calculer la différentielle de diverses opérations sur les espaces vectoriels. Par souci de concision, nous n'estimerons pas à partir de quel rayon la conclusion du Lemme 2.2.4 est vérifiée (ceci pourrait être fait en utilisant la Proposition 2.3.16) et nous supposons aussi que $\text{char}(K) = 0$.

Images directes

Soit E et F deux K -espaces vectoriels de dimension n et m , respectivement. Soit d un entier de $\llbracket 0, n \rrbracket$. Nous nous intéressons à l'application image directe, DI , définie sur $\mathcal{M} = \text{Hom}(E, F) \times \text{Grass}(E, d)$ et qui envoie un couple (f, V) sur $f(V)$. Puisque la dimension de $f(V)$ peut varier, l'application DI ne prend pas ses valeurs dans une grassmannienne fixe. Nous choisissons donc de stratifier \mathcal{M} de la manière suivante : pour tout entier $r \in \llbracket 0, d \rrbracket$, soit $\mathcal{M}_r \subset \mathcal{M}$ le sous-ensemble des couples (f, V) pour lesquels $f(V)$ est de dimension r . Les \mathcal{M}_r sont localement fermés dans \mathcal{M} et sont donc des sous-variétés. De plus, DI induit des fonctions différentiables $\text{DI}_r : \mathcal{M}_r \rightarrow \text{Grass}(F, r)$.

Nous souhaitons différencier DI_r en un point $(f, V) \in \mathcal{M}_r$. Pour cela, nous utilisons la première description des grassmanniennes donnée précédemment : nous voyons les points de $\text{Grass}(E, d)$ (resp. $\text{Grass}(F, d)$) comme des inclusions $V \hookrightarrow E$ (resp. $W \hookrightarrow F$) modulo l'action de $\text{GL}(V)$ (resp. $\text{GL}(W)$). Le point $V \in \text{Grass}(E, d)$ est alors représenté par l'injection canonique $v : V \rightarrow E$ tandis qu'un représentant w de W satisfait $w \circ \varphi = f \circ v$ où $\varphi : V \rightarrow W$ est l'application linéaire induite par f . Les relations précédentes sont encore vérifiées si (f, v) est remplacé par un couple $(f', v') \in \mathcal{M}_r$ suffisamment proche de (f, v) . En différenciant et en passant au quotient nous obtenons tout d'abord que l'espace tangent de \mathcal{M}_r en (f, v) est formé de couples $(df, dv) \in \text{Hom}(E, F) \times \text{Hom}(V, E/V)$ tels que

$$d\tilde{w} = ((df \circ v + f \circ dv) \bmod W) \in \text{Hom}(V, F/W)$$

se factorise à travers φ (i.e. s'annule sur $\ker \varphi = V \cap \ker f$). Ensuite, la différentielle de DI_r en (f, V) est l'application linéaire envoyant (df, dv) comme précédemment sur le seul élément $dw \in \text{Hom}(W, F/W)$ tel que $dw \circ \varphi = d\tilde{w}$.

Images réciproques

Nous considérons maintenant l'application image réciproque Π (pour *inverse image*) envoyant un couple $(f, W) \in \mathcal{W} = \text{Hom}(E, F) \times \text{Grass}(F, d)$ sur $f^{-1}(W)$. Comme précédemment, cette application ne prend pas ses valeurs dans une seule grassmannienne et nous devons stratifier \mathcal{W} afin d'obtenir des applications différentiables. Pour chaque entier $r \in [0, n]$, nous introduisons la sous-variété \mathcal{W}_s de \mathcal{W} formée des couples (f, W) tels que $\dim f^{-1}(W) = s$. Pour tout s , Π induit une fonction continue $\Pi_s : \mathcal{W}_s \rightarrow \text{Grass}(E, s)$. Prenons $(f, W) \in \mathcal{W}_s$. Soit $V = f^{-1}(W)$ et notons par $w : F \rightarrow F/W$ la projection canonique. De la même manière que pour les images directes, nous pouvons montrer que l'espace tangent à \mathcal{W}_s en un point $(f, W) \in \mathcal{W}_s$ est le sous-espace de $\text{Hom}(E, F) \times \text{Hom}(W, F/W)$ formé des couples (df, dw) tels que $d\tilde{v} = (w \circ df + dw \circ f)|_V$ se factorise par l'application linéaire $\varphi : E/V \rightarrow F/W$ induite par f . En outre, Π_s est différentiable en (f, W) et sa différentielle est l'application linéaire qui associe à (df, dw) comme précédemment l'unique élément $dv \in \text{Hom}(V, E/V)$ satisfaisant $\varphi \circ dv = d\tilde{v}$.

Images directes et images réciproques sont liés par la dualité de la manière suivante : si $f : E \rightarrow F$ est une application linéaire et W est un sous-espace de F , alors $f^*(W^\perp) = f^{-1}(W)^\perp$. Nous pouvons en déduire les différentielles de DI_r par celles de Π_s et réciproquement.

Remarque 3.3.1. Un cas particulier intéressant d'images réciproques est, bien sûr, celui des noyaux.

Sommes et intersections

Soit d_1 et d_2 deux entiers positifs. Nous considérons la fonction Σ définie sur la variété $\mathcal{C} = \text{Grass}(E, d_1) \times \text{Grass}(E, d_2)$ par $\Sigma(V_1, V_2) = V_1 + V_2$. Comme précédemment, afin d'étudier Σ , nous stratifions \mathcal{C} selon la dimension de la somme : pour chaque entier $d \in \llbracket 0, d_1 + d_2 \rrbracket$, nous définissons \mathcal{C}_d comme la sous-variété de \mathcal{C} formée des couples (V_1, V_2) tels que $\dim(V_1 + V_2) = d$. Nous obtenons ainsi une application bien définie $\mathcal{C}_d \rightarrow \text{Grass}(E, d)$ dont la différentielle peut être calculée comme précédemment. L'espace tangent de \mathcal{C}_d en un point (V_1, V_2) est formé des couples $(dv_1, dv_2) \in \text{Hom}(V_1, E/V_1) \times \text{Hom}(V_2, E/V_2)$ tels que $dv_1 \equiv dv_2 \pmod{V_1 + V_2}$ sur l'intersection $V_1 \cap V_2$ et la différentielle de Σ_r en (V_1, V_2) envoie (dv_1, dv_2) sur $dv \in \text{Hom}(V, E/V)$ (avec $V = V_1 + V_2$) défini par $dv(v_1 + v_2) = dv_1(v_1) + dv_2(v_2)$ ($v_1 \in V_1, v_2 \in V_2$).

En utilisant la dualité, nous pouvons obtenir un résultat similaire pour l'application $(V_1, V_2) \mapsto V_1 \cap V_2$.

3.3.3. Implémentations et expériences

Représentations standard d'espaces vectoriels

Nous avons vu que l'on représente les sous-espaces de K^n en utilisant les cartes $\mathcal{U}_{V_I, V_{I^c}}$ (où I est un sous-ensemble de $\{1, \dots, n\}$) introduites plus haut. Plus concrètement, un sous-espace $V \subset K^n$ est représenté par l'espace vectoriel engendré par les lignes de la matrice G_V , avec la propriété supplémentaire suivante : il existe un $I \subset \{1, \dots, n\}$ telle que la sous-matrice de G_V obtenue en gardant seulement les lignes dont l'indice est dans I soit la matrice identité. Nous rappelons qu'une telle représentation existe toujours, et dès que l'ensemble d'indices I est fixé, au plus un G_V satisfait la condition précédente. Étant donnée une famille de générateurs de V , il est possible de calculer G_V et I comme précédemment en effectuant un échelonnement en lignes. En choisissant le premier pivot n'est pas effectif, choisir le pivot de norme maximal fournit un algorithme plus stable.

La représentation duale

Bien sûr, il est aussi possible de travailler dans les cartes $\mathcal{U}_{V_I, V_{I^c}}^*$. Concrètement, ceci veut dire que nous représentons V comme le noyau à gauche d'une matrice H_V ayant la propriété suivante : il existe $I \subset \{1, \dots, n\}$ telle que la sous-matrice de H_V obtenue en *supprimant* les lignes ayant leur indice dans I soit la matrice identité. Comme précédemment, nous pouvons alors calculer I et H_V en faisant une réduction sur les colonnes.

Notons que changer de représentation est rapide et stable. En effet, si $I = \{1, \dots, d\}$ avec $d = \dim V$ et I_d la matrice identité de taille d , la matrice G_V est de la forme $(I_d \ G'_V)$. On peut représenter V avec le même I et la matrice $H_V = \begin{pmatrix} -G'_V \\ I_{n-d} \end{pmatrix}$. Une formule similaire peut être écrite pour tout I .

Opérations sur les espaces vectoriels

La première représentation que nous avons donnée est adaptée aux calculs d'images directes et de sommes. Par exemple, pour calculer $f(V)$, nous pouvons appliquer f aux lignes de G_V , pour obtenir une famille de générateurs de $f(V)$, et ensuite faire un échelonnement en lignes. De manière duale, la seconde représentation fonctionne bien pour calculer des images réciproques, par exemple des noyaux et des intersections. Puisque passer d'une représentation à l'autre est aisé, nous en déduisons directement des algorithmes adaptés utilisant chacune des représentations.

Quelques expériences

Considérons l'exemple du calcul donné par l'algorithme suivant :

Algorithme 3.3.2 : Opérations aléatoires sur des sous-espaces vectoriels

Entrée : Deux entiers n et N .

Sortie : L_n , obtenu par n opérations aléatoires à précision initiale $O(2N)$

début

Soit $L_0 = \langle (1 + O(2^N), O(2^N), O(2^N)) \rangle \subset \mathbb{Q}_2^3$;

pour $i = 0, \dots, n - 1$ **faire**

 Choisir aléatoirement $\alpha, \beta, \gamma, \delta \in M_{3,3}(\mathbb{Z}_2)$ avec précision grande devant N ;

 Calculer $L_{i+1} = (\alpha(L_i) + \beta(L_i)) \cap (\gamma(L_i) + \delta(L_i))$;

Retourner L_n ;

L'expression dite de grande précision devant N veut dire que α, β, γ et δ sont choisis à une précision suffisamment grande pour ne pas affecter la précision sur L_{i+1} . Avec le tableau en Figure 3.3 page 81, on voit que les pertes de précision lorsqu'on effectue l'Algorithme 3.3.2 avec des entrées n variées (l'entrée N est toujours choisie pour être suffisamment grande pour ne pas affecter le comportement de la précision). La colonne nommée *Coordonnée* correspond à la manière classique de suivre la précision. D'un autre côté, dans les deux dernières colonnes, la précision est suivie avec des réseaux. La colonne *Diffusée* donne la quantité de précision diffusée, écrite de manière à pouvoir

| n | Perte de précision moyenne | | |
|-----|----------------------------|----------|------------------|
| | Coordonnées méthode | Réseaux | |
| | | Projetée | Diffusée |
| 10 | 7,3 | 2,7 | $-2,4 \times 2$ |
| 20 | 14,8 | 5,5 | $-4,7 \times 2$ |
| 50 | 38,6 | 13,1 | $-12,0 \times 2$ |
| 100 | 78,1 | 26,5 | $-23,5 \times 2$ |

Résultats pour un échantillon de 1000 exécutions (avec $N \gg n$)

FIGURE 3.3. – Perte de précision dans l’Algorithme 3.3.2

être comparée au suivi par coordonnée. Le fait que seulement des nombres négatifs apparaissent veut dire que l’on gagne toujours de la précision dans ce modèle ! Finalement, la colonne nommée *Projetée* donne la perte de précision en projetant le réseau de précision sur ses coordonnées.

4. Précision en pratique : méthodes adaptatives

"Men that are trapped by the chains
of "maybe" can never reach their
dreams!"

Godot, *Phoenix Wright : Ace
Attorney : Trials and Tribulations*

"Give a man a fire and he's warm for
a day, but set fire to him and he's
warm for the rest of his life."

Terry Pratchett, *Jingo*

Le chapitre précédent, bien que présentant des bornes et des résultats explicites concernant la gestion de la précision p -adique n'explique pas comment gérer les calculs en pratique. Nous allons voir que les résultats précédents ne sont pas directement suffisants pour une approche effective. Cependant, nous proposons une méthode, dite "adaptative", pour atteindre la perte de précision définie dans le Lemme 2.2.4 de manière efficace. Il s'agit, là encore, d'un travail en commun avec Xavier Caruso et David Roe [CRV14].

Nous débutons en Section 4.1 par illustrer comment les calculs effectués en pratique ne reflètent pas nécessairement l'étude théorique effectuée dans le contexte de la précision différentielle. En conséquence, nous proposons en Section 4.2 ce que nous appelons une méthode adaptative pour atteindre en pratique le comportement de la précision prédit par l'étude différentielle de la précision. En Section 4.3, nous revenons sur les échecs traités en Section 4.1 pour montrer comment notre méthode a bien les performances souhaitées.

4.1. Échecs de certains calculs directs

Afin d'illustrer le fait que l'on n'atteint pas nécessairement naïvement les pertes de précision définies par le Lemme 2.2.4, nous revenons sur deux exemples traités en Section 1.4.

4.1.1. Retour sur la suite de SOMOS-4

Nous avons vu en Sous-Sous-Section 1.4.2 que, du fait que la suite de SOMOS-4 vérifie le phénomène de Laurent, connaître x_0, \dots, x_3 (inversibles) à précision $O(p^n)$ implique que tous les x_i sont définis à précision $O(p^n)$. Cependant, si nous écrivons naïvement dans Sage la formule de récurrence de la suite, $x_n = \frac{x_{n-3}x_{n-1} + x_{n-2}^2}{x_{n-4}}$, et que nous calculons les termes de la suite itérativement

4. Précision en pratique : méthodes adaptatives

avec cette formule, nous obtenons les résultats suivants :

$$\begin{aligned} x_0 &= 1 + O(5^{20}), \\ x_1 &= 1 + O(5^{20}), \\ x_2 &= 1 + O(5^{20}), \\ x_3 &= -1 + 5 + O(5^{20}), \\ x_4 &= 4 * 5 + \dots + O(5^{20}), \\ x_8 &= 4 + \dots + O(5^{19}), \\ x_{40} &= 4 + \dots + O(5^{13}). \end{aligned}$$

Remarquons que les quatre premiers termes sont bien pris comme étant inversibles, que tous les x_i obtenus sont bien dans \mathbb{Z}_p mais que certains, comme x_4 ou x_8 ne sont pas inversibles. Ces résultats correspondent avec l'étude naïve de pas à pas de la précision. En un certain sens, il semble que des gains de précisions, qui devraient compenser les divisions par p qui apparaissent dans la formule de récurrence, ne sont pas vus par Sage. Ainsi, les calculs directs ne sont pas suffisant pour atteindre la perte de précision intrinsèque.

Nous remarquons que l'algorithmique détendue n'apporte pas de changement à cet état de fait. En effet, une implémentation directe de la formule de récurrence de la suite SOMOS-4 en Mathemagix [vdHLM⁺12] nous montre que pour obtenir 20 chiffres sur x_{40} , Mathemagix en utilise 27 sur x_0, x_1, x_2 et x_3 , soit la même perte de précision induite de 7 chiffres.

4.1.2. Retour sur les arrangements

Revenons rapidement à l'exemple de la Sous-Sous-Section 1.4.2. Rappelons que nous souhaitons calculer tous les nombres d'arrangements A_n^k pour k fixé et n jusqu'à un certain $m \in \mathbb{N}$ donné, avec la contrainte supplémentaire de ne disposer des entiers qu'à précision $O(p^N)$ pour un p premier et un $N \in \mathbb{N}^*$ donné. Nous souhaitons obtenir alors ces nombres d'arrangements avec la meilleure précision possible. En conséquence, à partir de la formule $A_n^k = \frac{n!}{(n-k)!} = \prod_{i=n-k+1}^n i$, le calcul qui nous intéresse est $F_n(1, \dots, n)$ avec $F_n = \prod_{i=n-k+1}^n X_i \in \mathbb{Q}[X_1, \dots, X_n]$. Il est clair, en développant $F_n(1 + O(p^N), \dots, n + O(p^N))$ que ce produit est déterminé à précision $O(p^{N + \text{val}_p(A_n^k) - \max_i(\text{val}_p(i))})$ dès que $N > \max_i(\text{val}_p(i))$, et qu'il s'agit de la précision donnée par le premier ordre. C'est aussi ce que nous aurait donné une application de la Proposition 2.3.14.

Pour des raisons de temps de calcul, nous nous intéressons à l'application de la formule de récurrence $F_{n+1} = \frac{X_{n+1}}{X_{n-k+1}} F_n$, évaluée en les $i + O(p^N)$. Nous avons cependant vu que nous perdons de la précision lors de l'application de cette formule. Ainsi, ici encore, le calcul direct n'amène pas à la précision prévue théoriquement. Nous allons voir comment y remédier.

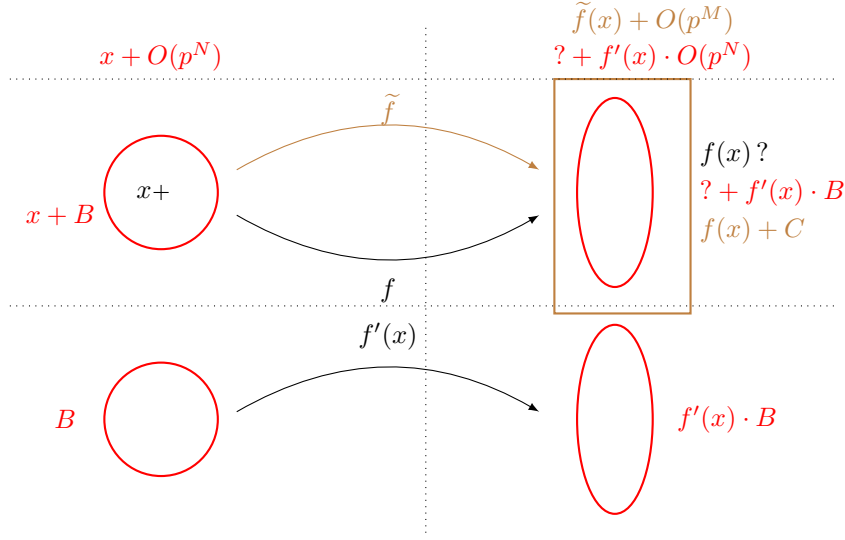
4.2. Une méthode adaptative

Malgré les échecs des calculs directs de la Section précédente, nous allons voir qu'il est possible de proposer une méthode pour atteindre la perte de précision prédite par le Lemme 2.2.4.

4.2.1. Illustration du problème

Tout d'abord, nous tentons de diagnostiquer ce qui est arrivé sur les exemples précédents. Ceci correspond à l'illustration suivante où nous nous intéressons au calcul de $f(x)$ connaissant seulement x à précision B^1 ainsi que le comportement de $f'(x)$. Le cadre est le suivant. Soit E, F deux K -espaces de Banach et $f : E \rightarrow F$ une application différentiable en $x \in E$. Supposons que $f'(x)$ est surjective et que B est un réseau du premier ordre pour f en x . Soit \tilde{f} une implémentation en machine de f . La situation qui nous intéresse est illustré par ce qui suit.

1. Connaître x à précision B veut dire connaître $x + B$ et B . On ne connaît pas x exactement, et n'importe quel point y de $x + B$ définit le même $y + B = x + B$.



En troisième ligne de cette illustration, nous avons l'étude théorique de la différentielle, qui nous dit comment la différentielle transforme la donnée de précision B , représentée par un cercle rouge, en $f'(x) \cdot B$, représenté par une ellipse rouge.

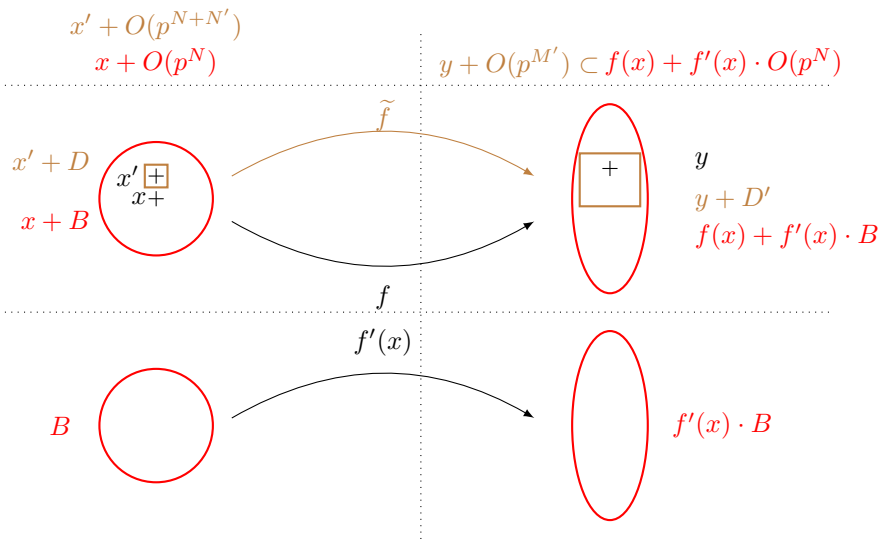
Cette étude théorique nous dit quelle est la précision sur $f(x)$ définie par $B : f'(x) \cdot B$ (au moins si B est assez petit). Le Lemme 2.2.4 nous dit en effet que $f(x + B) = f(x) + f'(x) \cdot B$. Néanmoins, et c'est ce qui est représenté en deuxième ligne, ceci ne nous dit pas où attacher cette donnée de précision $f'(x) \cdot B$ puisque $f(x)$ n'est bien sûr pas connu !

En marron est représenté ce que donnerait le calcul direct (par exemple donné par l'application de formules sur la multiplication ou la division) : il envoie $x + B$ sur le gros rectangle marron $f(x) + C$. Celui-ci peut contenir plusieurs $y + f'(x) \cdot B$ et ainsi, nous n'atteignons pas la précision attendue.

Enfin, la première ligne correspond simplement à l'analogie en dimension 1, sur \mathbb{Q}_p de la discussion précédente.

4.2.2. Illustration d'une solution

Nous allons maintenant illustrer la solution que nous proposons pour résoudre le problème précédent. Le contexte et les notations restent inchangés.



Là encore, la troisième ligne de l'illustration correspond à la gestion théorique de la précision par la différentielle de f . Nous expliquons sur la deuxième ligne notre méthode. Il s'agit d'ajouter arbitrairement de la précision autour d'un point de $x + B$, disons $x' + D$, et de calculer son image en machine, en utilisant les formules d'addition, multiplication, division. On obtient un certain $y + D'$ représenté à droite sur la deuxième ligne par un rectangle marron. Si D est assez petit et qu'ainsi

4. Précision en pratique : méthodes adaptatives

$D' \subset f'(x) \cdot B$, nous pouvons conclure. En effet, pour un point a et un réseau L , si $b \in a + L$, alors $a + L = b + L$, et en conséquence, si B est un réseau, alors $y + f'(x) \cdot B = f(x) + f'(x) \cdot B$. Ainsi, le calcul (numérique) de $y + D'$ combiné avec le calcul théorique de $f'(x) \cdot B$ sont suffisant dès que $D' \subset f'(x) \cdot B$.

En dimension 1, cf première ligne, ceci correspond à $y + O(p^{M'}) \subset f(x) + O(p^{N'})$ et ainsi $y + O(p^{N'}) = f(x) + O(p^{N'})$ (avec $N' = N + \text{val}(f'(x))$). Géométriquement ceci correspond à la remarque classique que sur \mathbb{Q}_p , tout point d'une boule est un centre de cette boule.

Nous remarquons que le $x' + D$ n'a pas besoin de vérifier que $x \in x' + D$, ou autrement dit, il ne correspond pas à une meilleure approximation du x initial dont on cherche à calculer l'image. Il doit néanmoins vérifier que $x' + D \subset x + B$.

Avec la discussion précédente, nous pouvons énoncer notre méthode de la manière suivante :

Lemme 4.2.1. *Soit E, F deux K -espaces de Banach et $f : E \rightarrow F$ une application différentiable en $x \in E$. Supposons que $f'(x)$ est surjective et que B est un réseau du premier ordre pour f en x . Soit \tilde{f} une implémentation en machine de f . La méthode suivante permet de déterminer $f(x + B)$ grâce à \tilde{f} et $f'(x) \cdot B$, et ce, même si $\tilde{f}(x + B) \supsetneq f(x + B)$.*

1. Déterminer $f'(x) \cdot B$.
2. Prendre $x' + D \subset x + B$ avec D assez petit.
3. Calculer $\tilde{f}(x' + D) = y + D'$ de manière directe.
4. Si $D' \subset f'(x) \cdot B$ (i.e. D était assez petit) alors $y + f'(x) \cdot B = f(x) + f'(x) \cdot B$.

Bien sûr, il est possible d'appliquer cette méthode en ne connaissant seulement qu'un C tel que $f'(x) \cdot B \subset C$ et de même, seulement une majoration de $\tilde{f}(x' + D)$.

4.3. Applications

Nous allons maintenant appliquer explicitement cette méthode sur nos deux exemples précédents que sont la suite de SOMOS-4 et les nombres d'arrangement.

4.3.1. Conclusion sur SOMOS-4

Présentation de l'algorithme stabilisé

Nous avons vu en Sous-Sous-Section 1.4.2 que la suite de SOMOS-4 satisfait le phénomène de Laurent et qu'ainsi, si $u_0, u_1, u_2, u_3 \in \mathbb{Z}_p^\times$ sont connus à précision $O(p^N)$, alors tous ses termes $u_n \in \mathbb{Z}_p$ pour $n \in \mathbb{N}$ sont connus à précision $O(p^N)$. Nous proposons avec l'Algorithme 4.3.1 un moyen d'atteindre cette précision en pratique, modulo une hypothèse raisonnable sur la valuation des coefficients u_n .

Algorithme 4.3.1 : SOMOS(a, b, c, d, n, N)

Entrées : a, b, c, d — les quatres termes initiaux de la suite SOMOS-4 $(u_n)_{n \geq 0}$

Entrées : n, N — deux entiers

Hypothèse : a, b, c et d sont dans \mathbb{Z}_p^\times et connus à précision $O(p^N)$.

Hypothèse : Aucun des u_i ($0 \leq i \leq n$) n'est de valuation plus grande que N .

Sortie : u_n à précision $O(p^N)$

début

```

    prec ← N;
    pour i de 1 à n − 3 faire
        prec ← prec + v_p(bd + c^2);
        Remonter b, c et d arbitrairement à la précision O(p^prec);
        prec ← prec − 2 v_p(a);
        e ← (bd + c^2) / a; // e est connu à précision O(p^prec)
        a, b, c, d ← b + O(p^prec), c + O(p^prec), d + O(p^prec), e;
    retourner d + O(p^N);

```

La ligne débutant par **Remonter** correspond, dans le contexte de la Section 4.2 à la création du rectangle marron $x' + C \subset x + B$. Deux difficultés sont présentes pour montrer la correction de l'algorithme. La première est de montrer que l'on a bien $x' + C \subset x + B$. Ceci n'est en effet pas évident puisque le réseau B n'est pas connu explicitement. La seconde est de montrer que le $f(x' + C) = y + C'$ est bien tel que $C' \subset f'(x) \cdot B$.

Correction de l'algorithme

Nous prouvons maintenant que l'algorithme est correct. Pour cela, nous allons noter notre fonction de récursion $f : \mathbb{Q}_p^\times \times \mathbb{Q}_p^3 \rightarrow \mathbb{Q}_p^4$ définie par $f(a, b, c, d) = (b, c, d, \frac{bd+c^2}{a})$. Pour tout i , nous avons $(u_i, u_{i+1}, u_{i+2}, u_{i+3}) = f_i(u_0, u_1, u_2, u_3)$ où $f_i = f \circ \dots \circ f$ (i fois). Clairement, f est différentiable sur $\mathbb{Q}_p^\times \times \mathbb{Q}_p^3$ et sa différentielle s'écrit (dans la base canonique de \mathbb{Q}_p^4) selon la matrice jacobienne suivante :

$$D(a, b, c, d) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{bd+c^2}{a^2} & \frac{d}{a} & \frac{2c}{a} & \frac{b}{a} \end{pmatrix}$$

dont le déterminant est $\frac{bd+c^2}{a^2}$.

Ainsi, si le $(i+4)$ -ème terme de la suite de SOMOS-4 est bien défini, l'application f_i est différentiable en (u_0, u_1, u_2, u_3) et sa différentielle $\varphi_i = f'_i(u_0, u_1, u_2, u_3)$ est donnée par la matrice $D_i = D(u_{i-1}, u_i, u_{i+1}, u_{i+2}) \cdots D(u_1, u_2, u_3, u_4) \cdot D(u_0, u_1, u_2, u_3)$. Grâce au phénomène de Laurent, nous savons que les coefficients de D_i sont dans $\mathbb{Z}[u_0^{\pm 1}, u_1^{\pm 1}, u_2^{\pm 1}, u_3^{\pm 1}]$, est donc φ_i stabilise le réseau \mathbb{Z}_p^4 . Nous allons maintenant prouver par récurrence sur i (pour $i \leq n-3$) que, à la fin du i -ème passage dans la boucle **pour** de l'algorithme, nous avons $\text{prec} = N + v_p(\det D_i)$ et

$$(a, b, c, d) \equiv (u_i, u_{i+1}, u_{i+2}, u_{i+3}) \pmod{p^N \varphi_i(\mathbb{Z}_p^4)}. \quad (4.1)$$

L'initialisation est évidente. Montrons l'hérédité en supposant le résultat vrai pour le $i-1$ -ème passage dans la boucle (*i.e.* le calcul de u_{i+2}) et montrons-le pour le i -ème, avec $i < n-3$. Tout d'abord, l'hypothèse que $\text{val}(u_j) \leq N$ pour tout $j \leq n$ nous garantit que la division par a dans cette i -ème boucle n'introduira pas une erreur de division par zéro (ou plutôt par $O(p^{\text{prec}})$). Maintenant, la démonstration du premier point, sur prec , est facile. Puisque $D_i = D(u_{i-1}, u_i, u_{i+1}, u_{i+2}) D_{i-1}$, nous en déduisons que $\det D_i = \det D_{i-1} \cdot \frac{u_{i+3}}{u_{i-1}}$. Comme entre le début et la fin de la boucle i , prec reçoit $\text{prec} + \text{val}(\frac{u_{i+3}}{u_{i-1}})$, le résultat est clair. Montrons maintenant que l'équation (4.1) est vérifiée. Pour éviter toute confusion, nous noterons par a', b', c', d' et prec' les valeurs prises par a, b, c, d et prec , respectivement, au *début* du i -ème passage dans la boucle **pour**, et nous conserverons les notations a, b, c, d et prec pour leur valeur à la sortie de cette boucle. Par hypothèse de récurrence, ou par définition si $i = 1$ (avec $\varphi_0 = Id$), nous avons :

$$(a', b', c', d') \equiv (u_{i-1}, u_i, u_{i+1}, u_{i+2}) \pmod{p^N \varphi_{i-1}(\mathbb{Z}_p^4)}. \quad (4.2)$$

De plus, nous avons vu que le déterminant de φ_{i-1} est de valuation $\text{val}(D_i) = \text{prec}' - N$. $D_i(u_0, \dots, u_3)$ est une matrice à coefficient dans \mathbb{Z}_p , donc (quitte à utiliser la formule de la comatrice), nous en déduisons que $p^{\text{prec}'} \mathbb{Z}_p^4$ est inclus dans $p^N \varphi_{i-1}(\mathbb{Z}_p^4)$. En conséquence, l'équation (4.2) reste vraie si a', b', c' and d' sont remplacés par toute autre valeur qui leur serait congruente modulo $p^{\text{prec}'}$. En particulier cette équation reste vraie si a', b', c' et d' prennent les valeurs de a, b, c et d après l'exécution de la ligne commençant par **Remonter**. Ceci correspond dans le contexte précédent, à avoir $x' + C \subset x + B$. En appliquant le Lemme 2.2.4 et la Proposition 2.3.8 à φ_{i-1} et φ_i (au point (u_0, u_1, u_2, u_3)), nous obtenons :

$$f((u_{i-1}, u_i, u_{i+1}, u_{i+2}) + p^N \varphi_{i-1}(\mathbb{Z}_p^4)) = (u_i, u_{i+1}, u_{i+2}, u_{i+3}) + p^N \varphi_i(\mathbb{Z}_p^4).$$

En effet, par définition et dérivée de la composée, $\varphi_i = f'(f_{i-1})\varphi_{i-1}$. Par la discussion précédente, cette équation implique en particulier que $f(a', b', c', d')$ appartienne à $(u_i, u_{i+1}, u_{i+2}, u_{i+3}) + p^N \varphi_i(\mathbb{Z}_p^4)$. Nous pouvons alors conclure en remarquant que $(a, b, c, d) \equiv f(a', b', c', d') \pmod{p^{\text{prec}} \mathbb{Z}_p^4}$ par construction et qu'à nouveau $p^{\text{prec}} \mathbb{Z}_p^4 \subset p^N \varphi_i(\mathbb{Z}_p^4)$.

4. Précision en pratique : méthodes adaptatives

Finalement, l'équation (4.1) appliquée pour $i = n - 3$ avec le fait que φ_i stabilise \mathbb{Z}_p^4 implique que, lorsque l'on sort de la boucle, la valeur de d est congrue à u_n modulo p^N . Remarquons en particulier que dans le contexte précédent, ceci correspondait à montrer que $C' \subset f'(x) \cdot B$. Pour conclure, notre algorithme retourne bien la bonne valeur.

Quelques mots sur la complexité

Nous achevons notre étude en remarquant que l'Algorithme 4.3.1 effectue ses calculs à précision au plus $O(p^{N+v})$ où v est le maximum de la somme des valuations de cinq termes consécutifs parmi les n premiers de la suite de SOMOS-4 considérée. En pratique, des exemples aléatoires montrent que la valeur de v semble varier comme $c \cdot \log n$ où c est une constante. En supposant que nous utilisons un algorithme comme la transformée de Fourier rapide pour calculer les produits d'entiers, la complexité de l'Algorithme 4.3.1 est alors espérée être en $\tilde{O}(Nn)$ où la notation \tilde{O} signifie à facteur logarithmique près.

Nous pouvons comparer ce résultat avec la complexité d'un algorithme plus naïf consistant à augmenter la précision sur les termes initiaux u_0, u_1, u_2, u_3 suffisamment, avec une borne donnée par une étude pas à pas, pour pouvoir effectuer des calculs directs des termes u_i de la suite. Ceci correspond *grosso modo* à ce que ferait de lui-même Mathemagix [vdHLM⁺12]. Dans ce contexte, la précision requise en début de calcul est $O(p^{N+v'})$ avec v' la somme des valuations des u_i pour i entre 0 et n . Des tests ont montré que v' se comporte comme $c' \cdot n \log n$ (avec c' une constante), ce qui amène une complexité en $\tilde{O}(Nn + n^2)$. Notre approche est alors intéressante lorsque n est grand comparé à N : sous cette hypothèse, nous gagnons, environ, un facteur n .

4.3.2. Conclusion sur les arrangements

En accord avec les méthodes développées précédemment, nous proposons un algorithme pour calculer des approximations des nombres d'arrangements à précision $O(p^N)$. Pour cela, si N est fixé et si $k \in \mathbb{N}$, nous notons \bar{k} pour l'entier dans $\llbracket 0, p^N - 1 \rrbracket$ tel que $k \equiv \bar{k} \pmod{p^N}$.

L'idée de l'algorithme est alors d'adapter la méthode adaptative. Dans ce cas, $k + O(p^N)$ correspond à ce que nous avons noté $x + B$. Nous avons bien $\bar{k} \in x + B$ et nous allons travailler avec des $\bar{k} + O(p^{N+N'})$ qui tiendront lieu de $x' + D$. Comme nous avons déjà estimé $f'(x) \cdot B$, cela sera suffisant pour conclure.2

Algorithme 4.3.2 : Liste des arrangements, stabilisé

Entrée : m, k, p, N des entiers naturels, avec p premier.

Sortie : La liste des A_n^k pour n de 0 à m , à une précision au moins $O(p^N)$.

début

```

 $l \leftarrow [0, \dots, 0]$  (liste de  $k$  zéros) ;
 $u \leftarrow k! + O(p^{N+val_p(k!)})$  ;
l.Ajouter( $u$ ) ;
pour  $i \in \llbracket k, m \rrbracket$  faire
     $u \leftarrow \frac{u}{i+1-\bar{k}+O(p^{N+val_p(u)+val_p(i+1-\bar{k})})}$  ;
     $u \leftarrow (i+1 + O(p^{N+val_p(u)+val_p(i+1)})) \times u$  ;
    l.Ajouter( $u + O(p^{N+val_p(u)-\max_{j=i+2-k}^{i+1} val_p(\bar{j})})$ ) ;
Retourner  $l$  ;
```

Nous avons alors le résultat suivant :

Proposition 4.3.3. *Étant donnés m, k, p, N des entiers naturels, avec p premier, tels que $N > E(\log_p(m))$, alors l'algorithme 4.3.2 calcule la liste des $A_n^k + O(p^{N+val_p(u)-\max_{j=i+2-k}^{i+1} val_p(\bar{j})})$ pour n de 0 à m .*

Démonstration. En appliquant les formules directes des Propositions 1.1.1 et 1.1.2, il est clair que l'algorithme calcule les $F_n(\bar{1}, \dots, \bar{n})$ à la précision $O(p^{N+val_p(F_n(\bar{1}, \dots, \bar{n}))})$. Comme pour tout $k \in \llbracket 0, m \rrbracket$ nous avons $k - \bar{k} = O(p^N)$, nous en déduisons grâce à notre étude sur la différentielle en

Sous-Section 4.1.2 que $F_n(\bar{1}, \dots, \bar{n}) - F_n(1, \dots, n) = O(p^{N+val_p(A_n^k) - \max_{j=i+2-k}^{i+1} val(\bar{j})})$. Ceci conclut notre preuve. \square

Afin de pouvoir évaluer le temps de calcul, nous pouvons donner l'estimation suivante sur la valuation de A_n^k :

Lemme 4.3.4. *Nous avons pour tout $k \in \mathbb{N}^*$, $n \in \mathbb{N}^*$, $n > k$:*

$$val_p(A_n^k) \leq \frac{1}{p-1} (k-1 + (p-1) \log(n-k)).$$

Démonstration. Dans ce contexte, $val_p(A_n^k) = val_p(n!) - val_p((n-k)!)$. En appliquant la formule de Legendre, nous obtenons $val_p(A_n^k) = \frac{1}{p-1} (n - S_p(n) - (n-k) + S_p(n-k))$, avec pour tout $i \in \mathbb{N}$, $S_p(i)$ la somme des chiffres en base p de i . En majorant $S_p(n-k)$ par $(p-1) \log_p(n-k)$ et en minorant $S_p(n)$ par 1, nous obtenons le résultat. \square

Ceci permet de conclure concernant la complexité de l'Algorithme 4.3.2 :

Lemme 4.3.5. *L'Algorithme 4.3.2 a une complexité inférieure à celle de $2*m$ opérations à précision $N + \frac{1}{p-1} (k-1 + (p-1) \log(m-k))$.*

Nous remarquons que pour k grand, ceci est bien plus efficace que les km opérations à précision allant jusqu'à $N + \frac{1}{p-1} (k-1 + (p-1) \log(m-k))$ que demanderait une évaluation directe pour chaque n de la formule définissant A_n^k .

5. Un exemple complet : résolution d'équations différentielles

"You can't give up now ! There ain't no gettin' off of this train we on !"

Barret Wallace, *Final Fantasy 7*

"It's still magic even if you know how it's done."

Terry Pratchett, *A Hat Full of Sky*

Ce chapitre, qui clôt la première partie de cette thèse, est un travail en commun avec Pierre Lairez. Il a pour objectif de présenter un exemple traité du début à la fin selon les méthodes développées lors des chapitres précédents.

Pour cela, après une présentation du problème qui nous intéresse en Section 5.1, nous appliquons l'étude différentielle de celui-ci en Section 5.2. Il s'agit d'abord de faire l'étude au premier ordre puis d'appliquer les techniques analytiques pour obtenir un suivi quantitatif et au premier ordre de la précision.

En Section 5.3, nous appliquons une méthode adaptative pour atteindre effectivement le comportement de la précision décrit pour l'étude au premier ordre.

Enfin, en Section 5.4, nous discutons des applications pratiques des résultats développés lors des Sections précédentes.

5.1. Présentation du problème

5.1.1. Introduction et résultats principaux

Nous étudions ici le calcul d'une série formelle à coefficients p -adiques qui est solution d'une équation différentielle du premier ordre avec séparation des variables. Soit g , h et y des séries de $\mathbb{Z}_p[[t]]$ telles que $h(0) = 1$ et $g(0) \neq 0$. L'équation différentielle qui nous intéresse est la suivante :

$$\begin{cases} y' = g \cdot h(y), \\ y(0) = 0. \end{cases} \quad (\text{E})$$

Plus précisément, étant données des approximations de g et h , nous cherchons des approximations des coefficients de y , la solution de (E). L'équation (E) a toujours une solution dans $\mathbb{Q}_p[[t]]$, mais dans cette étude, nous nous restreindrons au cas où la solution est dans $\mathbb{Z}_p[[t]]$.¹ Le cas d'une condition initiale plus générale $y(0) = c$ se réduit au nôtre par le changement de fonction de $h(t)$ en $h(t + c)$. Un cas qui nous intéresse particulièrement est celui de l'équation linéaire $y' = g \cdot y$, avec $y(0) = 1$. Il peut s'écrire $y' = g \cdot (1 + y)$ avec la transformation $y \mapsto y - 1$, et ainsi, s'écrit bien sous la forme : $z' = g \cdot h(z)$ et $z(0) = 1$ (et $h(0) = 1$).

1. Nous pourrions bien sûr faire notre étude sur $K[[t]]$ où K est un $CDVF$ à précision finie de caractéristique nulle, et l'hypothèse serait l'existence d'une solution dans $O_K[[t]]$. Les résultats seraient identiques si la caractéristique du corps résiduel est non-nulle, mais par contre pourraient être grandement améliorés dans le cas contraire. Le premier cas contient en particulier le cas des extensions finies de \mathbb{Q}_p , dont font partie les extensions non-ramifiées qui intéressaient les auteurs de [GvdHL15].

5. Un exemple complet : résolution d'équations différentielles

Quoi qu'il en soit, la condition $y(0) = 0$ est choisie pour pouvoir assurer que la composition $h(y)$ a bien du sens lorsque h est une série formelle quelconque. Elle pourrait être généralisée à toute condition permettant la composition $h(y)$ et assurant que $h(y) \in \mathbb{Z}_p[[t]]^\times$, e.g. $h = t$ et $y(0) = 1$ comme plus haut dans le cas linéaire. Nous présentons ici plus précisément ce que nous entendons par approximation :

Définition 5.1.1. Soit $\varepsilon \in \mathbb{R}_+$ et $n \in \mathbb{N}$, nous appelons *approximation modulo* (ε, t^n) d'une série formelle $f \in \mathbb{Z}_p[[t]]$ toute série formelle $\tilde{f} \in \mathbb{Z}_p[[t]]$ telle que $|f_k - \tilde{f}_k| \leq \varepsilon$, pour tout $k < n$. Autrement dit, nous prenons comme type de précision sur $\mathbb{Z}_p[[t]]$ les réseaux (sur \mathbb{Z}_p) de la forme

$$(\varepsilon, t^n) = \bigoplus_{k=0}^{n-1} p^{E(\log_p(\varepsilon))} t^k \mathbb{Z}_p \oplus t^n \mathbb{Z}_p[[t]].$$

Comme suggéré par la forme de l'équation différentielle, l'opération de composition des séries formelles, $h(f)$, est importante. Soit $C_h(\varepsilon, n)$, le nombre d'opérations binaires nécessaire pour calculer une approximation modulo (ε, t^n) de $h(f)$ étant donnée une approximation modulo (ε, t^n) de la série formelle f . En général, h est facile à évaluer : il s'agit d'une fraction rationnelle dont le numérateur et le dénominateur sont de petit degré, ou une racine d'une telle fonction, de telle manière que $C_h(\varepsilon, n)$ est en $\mathcal{O}(M(n) |\log \varepsilon|)$. Dans tous les cas, Kedlaya et Umans ont prouvé dans [KU11] que

$$C_h(\varepsilon, n) = \mathcal{O}\left(n^{1+o(1)} |\log \varepsilon|^{1+o(1)}\right).$$

Bien sûr, $C_h(\varepsilon, n)$ est en $\Omega(n |\log \varepsilon|)$, ce qui correspond à la taille binaire de la sortie.

Notre résultat principal est alors le suivant :

Théorème 5.1.2. Soit α et n deux entiers positifs, et soit $\rho = p^{\frac{2}{1-\rho}}$. Soit $\varepsilon > 0$ un réel tel que $\varepsilon \leq \frac{\rho}{(n+1)^2}$. Alors, étant donné des approximations modulo (ε, t^n) de g et h , on peut calculer une approximation modulo $(n\varepsilon, t^{n+1})$ de la solution y de (E) en utilisant

$$\mathcal{O}(M(n) |\log \varepsilon| + C_h(\varepsilon, n))$$

opérations binaires.

Ce résultat a déjà été montré dans le cas linéaire dans [BGVPS05] grâce à une preuve astucieuse utilisant les sommes de Newton, ainsi que dans [GvdHL15] (où le cas des extensions non-ramifiées de \mathbb{Q}_p est aussi traité). Dans le cas non-linéaire, il était seulement connu dans [LS08] qu'une approximation modulo $(n^{\log_2 n} \varepsilon, t^n)$ pouvait être calculée à partir d'approximations modulo (ε, t^n) de g et h .

Notre preuve se fait en deux étapes. Tout d'abord, nous analysons la perte de précision intrinsèque dans la résolution de notre équation différentielle. Ceci veut dire que nous étudions quelle information il est possible d'obtenir sur y étant données des approximations de g et h . Nous utilisons le cadre et les méthodes développées dans le Chapitre 2. Dans un second temps, nous analysons un algorithme fondé sur une version adaptative d'une méthode de Newton pour calculer une approximation de y . Nous utilisons les méthodes du Chapitre 4 et montrons qu'il atteint la perte de précision optimale.

5.1.2. Applications

Sommes de Newton

Il s'agit du problème original, étudié dans [BGVPS05]. Soit $f \in \mathbb{Z}_p[t]$ un polynôme unitaire de degré d , et soit ν_n la n -ème somme de Newton de f . Si $\alpha_1, \dots, \alpha_d$ sont les racines de f dans $\overline{\mathbb{Q}_p}$, alors ν_n est la somme $\alpha_1^n + \dots + \alpha_d^n$. Nous avons $\nu_n \in \mathbb{Z}_p$. Le problème qui nous intéresse est d'obtenir f lorsque l'on connaît les sommes de Newton ν_0, \dots, ν_d . Soit g le polynôme $x^d f(1/x)$, et N_f la fonction génératrice $N_f(t) = \sum_{n \geq 0} \nu_{n+1} t^n$. Alors nous avons $g' = -N_f g$, et ainsi g est solution d'une équation différentielle linéaire du premier ordre. Ainsi, connaître une approximation modulo (ε, t^d) de N_f permet d'obtenir les coefficients de g (et ainsi de f) à la précision $d \times \varepsilon$.

Une application intéressante est le calcul sur \mathbb{F}_p de produits composés, qui est traitée dans [BGVPS05]. Soit f et g deux polynômes unitaires de $\mathbb{F}_p[t]$ de degrés d et e respectivement, et de

racines $(\alpha_i)_{1 \leq i \leq d}$ et $(\beta_j)_{1 \leq j \leq e}$ dans $\overline{\mathbb{F}_p}$ respectivement. Nous définissons le produit composé de f et g comme :

$$f \otimes g = \prod_{i,j} (t - \alpha_i \beta_j) = \operatorname{res}_y (y^p f(t/y), g(y)).$$

Il s'agit bien d'un polynôme de $\mathbb{F}_p[t]$, de degré $p+q$. Nous remarquons alors que $N_{f \otimes g}$ est le produit d'Hadamard des séries formelles N_f et N_g . Ceci nous donne une stratégie pour calculer efficacement $f \otimes g$:

1. Remonter f et g en des polynômes de $\mathbb{Z}_p[t]$, notés \bar{f} et \bar{g} . Le produit composé $\bar{f} \otimes \bar{g}$ est un polynôme de $\mathbb{Z}_p[t]$ et coïncide avec $f \otimes g$ modulo p .
2. Calculer des approximations modulo $((d+e)^{-1}, t^{d+e})$ de $N_{\bar{f}}$ et $N_{\bar{g}}$, puis, avec un produit d'Hadamard, une approximation modulo $((d+e)^{-1}, t^{d+e})$ de $N_{\bar{f} \otimes \bar{g}}$.
3. Calculer une approximation modulo $(1, t^{d+e+1})$ de $\bar{f} \otimes \bar{g}$ en utilisant le Théorème 5.1.2, et en déduire $f \otimes g$ en réduisant modulo p .

D'autres applications peuvent être trouvées dans [GvdHL15], notamment pour des calculs de transformées de Graeffe et la recherche de racines d'un polynôme sur un corps fini.

Notre travail n'améliore pas le résultat donné dans [BGVPS05] mais donne, nous l'espérons, une preuve plus simple du résultat, ainsi qu'une compréhension un peu plus profonde de la perte de précision dans la résolution de ces équations différentielles.

L'algorithme de Lercier-Sirvent

Pour calculer des isogénies normalisées entre courbes elliptiques, Bostan, Morain, Salvy et Schost [BMSS08] et Lercier et Sirvent [LS08] ont étudié l'équation différentielle

$$y'^2 = g \cdot h(y), \quad (5.1)$$

où g et h sont des séries dans $\mathbb{Z}_p[[t]]$, avec l'hypothèse supplémentaire d'avoir une solution à coefficient dans $\mathbb{Z}_p[[t]]$. Comme dans l'exemple précédent, il s'agit d'un problème de remontée puis descente entre \mathbb{F}_p et \mathbb{Z}_p . Cette équation se réécrit de manière équivalente $y' = \sqrt{g} \sqrt{h(y)}$, et si $p \neq 2$, les séries \sqrt{g} et \sqrt{h} sont encore dans $\mathbb{Z}_p[[t]]$, et ainsi nous pouvons appliquer le Théorème 5.1.2.

Nous obtenons alors que pour calculer n coefficients, à précision 1, de la solution de l'équation différentielle 5.1, il suffit de connaître g et h à précision $n^2 \rho$ (où $\rho = p^{\frac{2}{1-p}}$), ce qui améliore la borne $n^{\log_2 n}$ donnée dans [LS08].

5.2. Étude théorique, réseaux du premier ordre

Dans cette Section, nous appliquons les méthodes du Chapitre 2 pour étudier la perte de précision dans la résolution de l'équation différentielle (E). À cet effet, nous calculons d'abord la différentielle de l'application Y_n qui envoie $(g_{\leq n}, h_{\leq n})$ sur le n -ème coefficient de la solution de (E). La norme de cette différentielle nous donnera un premier résultat qualitatif sur la perte de précision dans le calcul de Y_n . Pour donner des résultats quantitatifs, nous estimons les normes des différentielles d'ordre supérieur de Y_n , puis appliquons le Corollaire 2.3.12.

5.2.1. La différentielle de Y_n

Notations

Soit $f = \sum_{m=0}^{+\infty} a_m t^m \in \mathbb{Q}_p[[t]]$ une série. Si $n \in \mathbb{N}$, nous noterons $f_{\leq n}$ pour le polynôme $\sum_{m=0}^n a_m t^m$ et $f_{< n}$ pour $\sum_{m=0}^{n-1} a_m t^m$. De manière similaire, $f_{\geq n} = \sum_{m=n}^{+\infty} a_m t^m$. Nous noterons aussi $[f]_{\leq n}$ pour le $n+1$ -uplet (a_0, \dots, a_n) des coefficients de $f_{\leq n}$, et de même pour $[f]_{< n}$. Enfin, $[f]_n = a_n$.

Définition 5.2.1. Soit $n \in \mathbb{N}$, nous définissons $Y_n : \begin{matrix} \mathbb{Q}_p[t]_{\leq n}^2 & \rightarrow & \mathbb{Q}_p \\ (g_{\leq n}, h_{\leq n}) & \mapsto & y_n, \end{matrix}$ ou y_n est le coefficient en t^n de la série y , unique solution de l'équation différentielle (E). Nous noterons alors

5. Un exemple complet : résolution d'équations différentielles

$Y_{\leq n}(g_{\leq n}, h_{\leq n}) = \sum_{m=0}^n Y_n(g_{\leq n}, h_{\leq n})t^m$ et nous avons bien sûr $y = \sum_{m=0}^{+\infty} Y_n(g_{\leq n}, h_{\leq n})t^m$. Nous définissons $Y_{< n}$ de manière analogue.

Calcul explicite

La première étape dans le suivi différentiel de la précision est le calcul de la dérivée de l'application dont la perte de précision nous intéresse. Dans notre cas, ceci revient à résoudre explicitement l'équation différentielle du premier ordre satisfaite par Y_n .

La différentielle de Y_n en un point $(g_{< n}, h_{< n})$ sera notée $d_{(g,h)}Y_n$. Maintenant, pour g et h dans $\mathbb{Q}_p[[t]]$, soit y l'unique solution de l'équation différentielle (E), et soit dY la série formelle

$$d_{(g,h)}Y \stackrel{\text{def}}{=} \sum_{n \geq 1} t^n d_{(g_{< n}, h_{< n})}Y_n.$$

Proposition 5.2.2. *Nous avons alors :*

$$d_{(g,h)}Y = h(y) \int \left(\frac{g}{h(y)} dh + dg \right)$$

Démonstration. Nous différencions $y' = g(t)h(y)$ pour obtenir au premier ordre :

$$y' + d_{(g,h)}Y' = g(t)h(y) + gh'(y)d_{(g,h)}Y + gdh(y) + h(y)dg.$$

En conséquence :

$$d_{(g,h)}Y' = gh'(y)d_{(g,h)}Y + gdh(y) + h(y)dg. \quad (5.2)$$

Cette dernière équation est une équation du premier ordre affine en $d_{(g,h)}Y$. Avec la condition initiale $d_{(g,h)}Y(0) = 0$, due à l'équation différentielle (E), l'équation différentielle 5.2 peut se résoudre directement dans $\mathbb{Q}_p[[t]]$ pour obtenir le résultat. Il suffit de vérifier que la formule donnée fonctionne. Une autre manière est de constater que formellement, et avec la méthode de variation de la constante, nous trouvons

$$d_{(g,h)}Y = e^{\int gh'(y_0)} \int e^{\int gh'(y)} (g(t)\delta h(y) + h(y))\delta g.$$

Or, en prenant la dérivée logarithmique de $y' = g(t)h(y)$, nous obtenons

$$gh'(y) = \frac{y''}{y'} - \frac{g'h(y)}{y'} = \frac{y''}{y'} - \frac{g'}{g}.$$

Le résultat s'en déduit directement. \square

Norme de la différentielle

Grâce à la forme explicite obtenue pour $d_{(g,h)}Y$ dans la Proposition 5.2.2, nous pouvons poursuivre notre étude selon le cadre de la précision différentielle avec l'étude de la norme des $d_{(g,h)}Y_{n+1}$. Ceci donnera une première idée qualitative au premier ordre du comportement de la perte de précision au voisinage de g et h définissant une solution à (E) dans $\mathbb{Z}_p[[t]]$. Le résultat est le suivant :

Proposition 5.2.3.

$$\|d_{(g,h)}Y_{n+1}\|_{\infty} \leq n + 1.$$

Démonstration. Puisque $y, y', g, h \in \mathbb{Z}_p[[t]]$ et $h(y) \in \mathbb{Z}_p^{\times}$, alors en développant la formule $d_{(g,h)}Y_{n+1} = \left(h(y) \int \frac{g}{h(y)} dh(y) + dg \right)_{n+1}$, le coefficient de $d_{(g,h)}Y_{n+1}$ dans dg_i ou dh_i peut être écrit

$$\frac{1}{p^{-E(\log_p(n+1))}} P([y_0]_{\leq n+1}, [g_0]_{\leq n+1}, [h_0]_{\leq n+1})$$

avec E la partie entière et $P \in \mathbb{Z}_{(p)}[y_{\leq n+1}, g_{\leq n+1}, h_{\leq n+1}]$ de degré total au plus $n + 1$. En effet, les nombres de valuation négative dans l'expression précédente ne peuvent venir que du fait que lorsqu'on intègre, $\int X^i = \frac{X^{i+1}}{i+1}$. En conséquence, la plus petite valuation qui puisse apparaître sur un terme de $d_{(g,h)}Y_{n+1}$ est $-E(\log_p(n+1))$, d'où le résultat. \square

5.2.2. Réseaux du premier ordre

Pour poursuivre l'application du cadre du Chapitre 2 l'étape suivante est de déterminer à quelle précision sur les entrées le premier ordre dirige la précision. À cet effet, nous suivons la Sous-Section 2.3.1, qui explique à quelle précision sur les entrées les termes issus des différentielles d'ordre supérieur sont absorbés par la précision au premier ordre.

La preuve est effectuée en trois étapes.

Étape 1 : Surjectivité et constante C

Nous allons prouver directement la surjectivité de $d_{(g,h)}Y_n$ et trouver une constante C comme dans le Corollaire 2.3.12.

Lemme 5.2.4. *Pour $C = 1$, nous avons $B_{\mathbb{Q}_p}(1) \subset d_{(g,h)}Y_n \cdot B_{\mathbb{Q}_p^{2n+2}}(C)$.*

Démonstration. Avec la Proposition 5.2.2, nous avons $d_{(g,h)}Y = h(y) \int \left(\frac{g}{h(y)} dh + dg \right)$. Ainsi, $d_{(g,h)}Y_n = \left(h(y) \int \left(\frac{g}{h(y)} dh + dg \right) \right)_{=n}$. Si $a \in B_{\mathbb{Q}_p}(1)$, soit $dh = 0$ et $dg = a(n-1)t^{n-1}$. Comme $h(0) = 1$ et $y(0) = 0$, nous en déduisons que dans ce cas, $d_{(g,h)}Y_n = a$. Or, $a(n-1) \in B_{\mathbb{Q}_p}(1)$. Le résultat s'en déduit directement. \square

En conséquence, nous prenons $C = 1$.

Étape 2 : Bornes sur les nomes des différentielles d'ordre supérieur

La deuxième étape consiste en le contrôle des termes du reste d'ordre 1 dans le développement de Taylor de Y_n en (g, h) . Pour cela, nous majorons directement la norme de la différentielle d'ordre k de Y_n en (g, h) définissant une solution dans $\mathbb{Z}_p[[t]]$.

Proposition 5.2.5. *Dans ces conditions, nous avons : $\|d_{(g,h)}^k Y_n\|_\infty \leq (n+1)^k$.*

$$\frac{1}{|k|!_p} \|d^k Y_n\|_\infty \leq e^{k \log p (\log_p(n+1) + \frac{1}{p-1})}.$$

Démonstration. Nous prouvons par récurrence sur k que pour tout α, β tels que $|\alpha| + |\beta| = k$, le coefficient de $d_{(g,h)}^k Y$ en $dg_{\leq n}^\alpha dh_{\leq n}^\beta$ est de la forme $\frac{1}{p^{kE(\log_p(n+1))}} Q_{\alpha,\beta}([y]_{\leq n+1}, [g]_{\leq n+1}, [h]_{\leq n+1})$ avec $Q_{\alpha,\beta} \in \mathbb{Z}_{(p)}[[y]_{\leq n+1}, [g]_{\leq n+1}, [h]_{\leq n+1}]$. Nous rappelons que $\mathbb{Z}_{(p)} = S^{-1}\mathbb{Z}$ avec $S = \mathbb{Z} \setminus (p)$ le localisé de \mathbb{Z} par rapport à son idéal premier (p) . Ce fait est clair pour $k = 1$, vue la Proposition 5.2.3. Maintenant, si α, β sont tels que $|\alpha| + |\beta| = k$ et que l'on note le coefficient de $d_{(g,h)}^k Y$ en $dg_{\leq n}^\alpha dh_{\leq n}^\beta$ par $\frac{1}{p^{kE(\log_p(n+1))}} Q_{\alpha,\beta}([y]_{\leq n+1}, [g]_{\leq n+1}, [h]_{\leq n+1})$ avec $Q_{\alpha,\beta} \in \mathbb{Z}_{(p)}[[y]_{\leq n+1}, [g]_{\leq n+1}, [h]_{\leq n+1}]$, nous pouvons regarder ce que l'on obtient en dérivant ce coefficient par rapport à g_i ou f_j . Nous pouvons nous restreindre à étudier un monôme de $Q_{\alpha,\beta}$, que nous noterons $[y]_{\leq n+1}^\gamma [g]_{\leq n+1}^{\gamma'} [h]_{\leq n+1}^{\gamma''}$. Rien de remarquable ne se passe lorsqu'on dérive l'un des deux derniers facteurs (en $[g]_{\leq n+1}$ ou $[h]_{\leq n+1}$) par rapport à un g_i ou un h_j . Par contre, lorsqu'on dérive $[y]_{\leq n+1}^\gamma$ par rapport à g_i , nous obtenons grâce à la Proposition 5.2.2, une somme de termes de la forme $\frac{\gamma_l}{p^{lE(\log_p(n+1))}} P_l([y_0]_{\leq n+1}, [g_0]_{\leq n+1}, [h_0]_{\leq n+1}) [y]_{\leq n+1}^{\gamma'} dg_i$, avec $P_l \in \mathbb{Z}_{(p)}[[y]_{\leq n+1}, [g]_{\leq n+1}, [h]_{\leq n+1}]$, $\gamma - \gamma' = (0, \dots, 0, 1, 0, \dots, 0)$ (1 en l -ème position), et $l \in \llbracket 0, n+1 \rrbracket$. Il en est de même lorsqu'on dérive par rapport à h_j .

Ainsi, nous en déduisons directement que si pour tout α, β tels que $|\alpha| + |\beta| = k$, le coefficient de $d_{(g,h)}^k Y$ en $dg_{\leq n}^\alpha dh_{\leq n}^\beta$ s'écrit $\frac{1}{p^{kE(\log_p(n+1))}} Q_{\alpha,\beta}([y]_{\leq n+1}, [g]_{\leq n+1}, [h]_{\leq n+1})$ avec $Q_{\alpha,\beta} \in \mathbb{Z}_{(p)}[[y]_{\leq n+1}, [g]_{\leq n+1}, [h]_{\leq n+1}]$, alors pour tout α', β' tels que $|\alpha'| + |\beta'| = k+1$, le coefficient de $d_{(g,h)}^{k+1} Y$ en $dg_{\leq n}^{\alpha'} dh_{\leq n}^{\beta'}$ s'écrit $\frac{1}{p^{(k+1)E(\log_p(n+1))}} Q_{\alpha',\beta'}([y]_{\leq n+1}, [g]_{\leq n+1}, [h]_{\leq n+1})$ avec

$$Q_{\alpha',\beta'} \in \mathbb{Z}_{(p)}[[y]_{\leq n+1}, [g]_{\leq n+1}, [h]_{\leq n+1}],$$

ce qui clôt la récurrence.

La première inégalité suit directement de ce résultat.

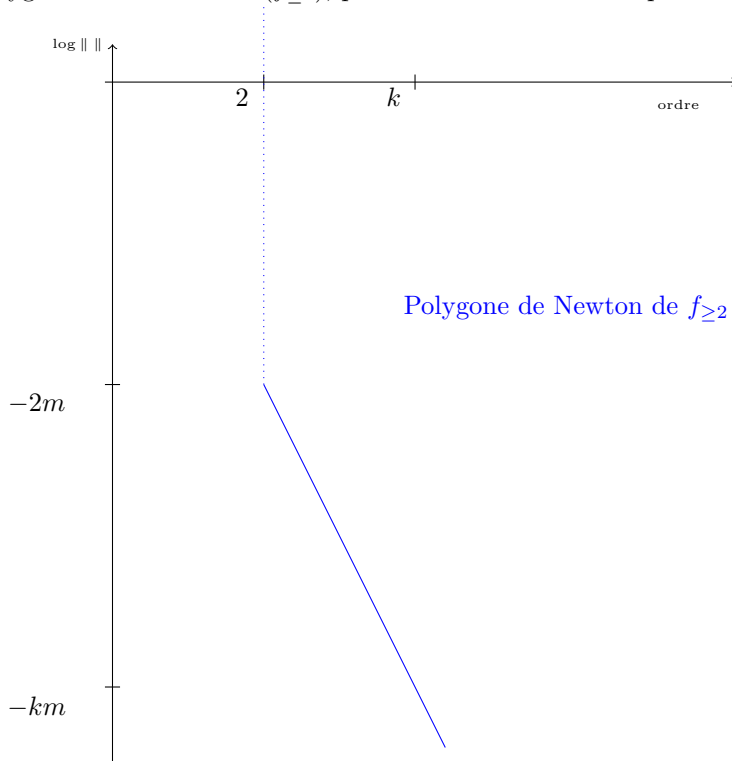
Pour la seconde formule, il suffit d'appliquer la formule de Legendre. Pour rappel, $val_p(n!) = \frac{n-S_p(n)}{p-1}$ où S_p est la somme des chiffres de n écrit en base p . \square

Étape 3 : polygones de Newton

Dans cette dernière étape, nous étudions les polygones de Newton définis à partir des normes des différentielles d'ordre supérieur de Y_n . Nous avons alors le lemme suivant :

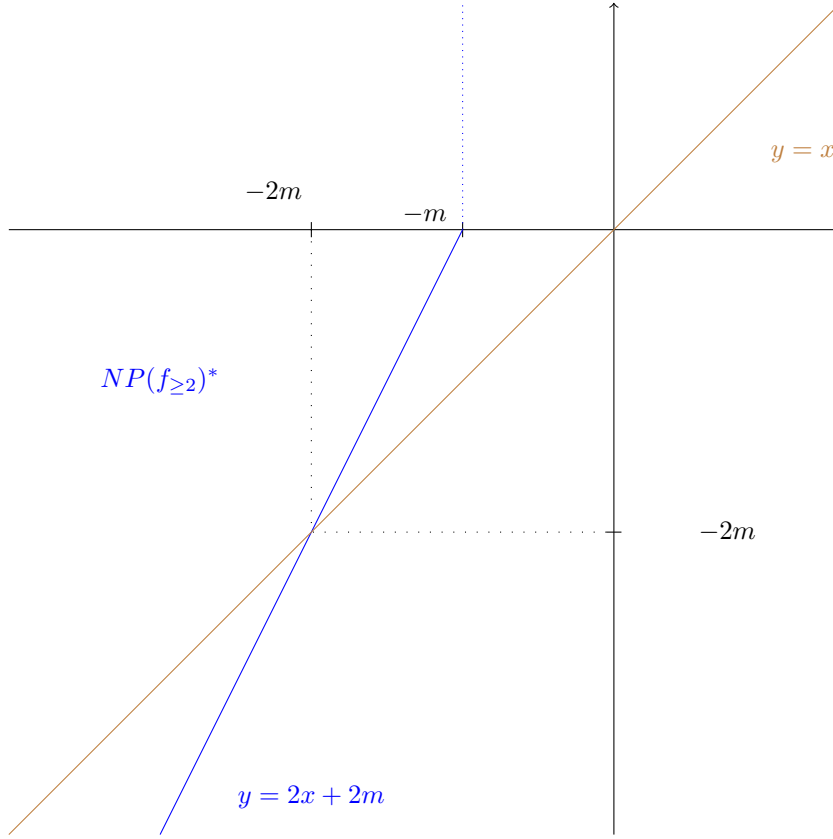
Lemme 5.2.6. *Soit $m = \log p \left(\log_p(n+1) + \frac{1}{p-1} \right)$. Alors $NP((Y_n)_{\geq 2})^*(\nu) < \nu$ dès que $\nu < -2m$.*

Démonstration. D'après la Proposition 5.2.5, la droite donnée par $y = -xm$ se trouve sous le polygone de Newton $NP(f_{\geq 2})$, pour $2 \leq x$. Nous avons représenté cette situation ici :



En conséquence, en utilisant que le passage à la transformée de Legendre est décroissant, $NP(f_{\geq 2})^*$ se trouve au-dessous de la droite donnée par $y = 2x + 2m$ pour $x \leq -1$.

Les droites d'équation $y = x$ et $y = 2x + 2m$ s'intersectent en $x = -2m$. Nous avons bien $-2m < -1$. En conséquence, pour tout $\nu < -2m$, $NP((Y_n)_{\geq 2})^*(\nu) < \nu$, ce que nous souhaitons montrer. La situation est représentée ici :



□

5.2.3. Conclusion sur la précision

Nous pouvons maintenant conclure et énoncer le résultat suivant sur le comportement de la précision sur $Y_n(g_{\leq n}, h_{\leq n})$ étant données des approximations de g et h :

Proposition 5.2.7. *Soit $\varepsilon < \frac{1}{(n+1)^2}$, alors :*

$$Y_n(g_{\leq n} + O(\varepsilon), h_{\leq n} + O(\varepsilon)) = y_n + Y'_n(g_{\leq n}, h_{\leq n}) \cdot [O(\varepsilon)],$$

Démonstration. Vus les lemmes 5.2.4 et 5.2.6, il nous suffit d'appliquer directement le Corollaire 2.3.12 pour conclure concernant la première partie de la proposition. □

5.3. Application effective, atteindre la borne

D'un premier abord, notre résultat en Proposition 5.2.7 sur la perte de précision semble seulement théorique. Cependant, nous montrons dans cette Section qu'il est possible d'atteindre effectivement la précision énoncée en stabilisant par les méthodes vues dans le Chapitre 4 une itération de Newton.

5.3.1. L'itération de Newton naïve ne suffit pas

Étant donnés $g, h \in \mathbb{Q}[[t]]$, une idée naturelle pour résoudre l'équation (E) dans $\mathbb{Q}[[t]]$ au lieu de $\mathbb{Q}_p[[t]]$ est d'appliquer un algorithme de Newton-Hensel, dont l'itération est donné par le lemme suivant :

Lemme 5.3.1. *L'itération de Newton suivante produit une suite convergeant quadratiquement (dans $\mathbb{Q}[[t]]$) vers la solution de (E) :*

5. Un exemple complet : résolution d'équations différentielles

$$u_0 = t \quad (5.3)$$

$$u_{l+1} = u_l + h(u_l) \int \frac{1}{h(u_l)} (g(t)h(u_l) - u'_l) \mod t^{2^{l+1}+1}, \quad (5.4)$$

Nous avons pour tout l , $g(t)h(u_l) = u'_l \mod t^{2^l}$.

Démonstration. Nous montrons le résultat suivant par récurrence sur l : pour tout $l \in \mathbb{N}$, $u_l \in \mathbb{Q}[[t]]$ est bien défini, $u_l = u_0 \mod t$ et $g(t)h(u_l) = u'_l \mod t^{2^l}$. Pour $l = 0$, le résultat est clair. Maintenant, si la propriété est vraie pour un certain $l \in \mathbb{N}$, montrons-là pour $l + 1$. Tout d'abord, comme $u_l = u_0 \mod t$, $u_l(0) = 0$ et comme $h(0) = 1$, $h(u_l)$ est bien inversible dans $\mathbb{Q}[[t]]$. En conséquence, $u_{l+1} \in \mathbb{Q}[[t]]$ est bien défini. Comme $g(t)h(u_l) = u'_l \mod t^{2^l}$, il est clair que $u_{l+1} = u_l \mod t$ et donc $u_{l+1} = u_0 \mod t$.

Maintenant, nous pouvons dériver l'expression $u_{l+1} = u_l + h(u_l) \int \frac{1}{h(u_l)} (g(t)h(u_l) - u'_l) \mod t^{2^{l+1}+1}$. Elle nous donne :

$$u'_{l+1} = u'_l + gh(u_l) - u'_l + u'_l h'(u_l) \int \frac{1}{h(u_l)} (gh(u_l) - u'_l) + O(t^{2^{l+1}}).$$

Ainsi, $u'_{l+1} = gh(u_l) + u'_l h'(u_l) \int \frac{1}{h(u_l)} (gh(u'_l) - u'_l) + O(t^{2^{l+1}})$. Or, comme $gh(u_l) - u'_l = O(t^{2^l})$, nous pouvons appliquer la formule de Taylor pour h en u_l pour obtenir :

$$gh(u_{l+1}) = gh(u_l) + gh'(u_l) \times h(u_l) \int \frac{1}{h(u_l)} (gh(u_l) - u'_l) + O(t^{2^{l+1}}).$$

En conséquence, nous avons bien $u'_{l+1} = gh(u_{l+1}) + O(t^{2^{l+1}})$. Le résultat est donc prouvé. \square

Cependant, si nous appliquons naïvement sur $\mathbb{Q}_p[[t]]$ cette itération, nous perdrons $E(\log_p(2^{l+1}))$ chiffres de précision à chaque itération du fait de l'intégration. Ceci amènerait une perte totale pour le calcul de y_n en $O(n^{\log_2 n})$, alors que la perte de précision théorique vue en Proposition 5.2.7 est en $O(n)$.

5.3.2. Stabilisation de la méthode de Newton-Hensel : une méthode adaptative

Nous proposons une stabilisation de la méthode de Newton-Hensel que nous avons présentée en utilisant une méthode adaptative, selon les idées du Chapitre 4.

Présentation de l'algorithme

Nous commençons par présenter l'algorithme stabilisé que nous proposons. L'idée principale est d'augmenter arbitrairement la précision sur u et g avant d'effectuer l'itération de Newton, de manière à ce que les chiffres de précision p -adique additionnels que nous ajoutons agissent comme tampon par rapport à la précision : ils absorbent la perte de précision due à l'itération de Newton, et préservent les chiffres significatifs connus. Nous supposons que h est connu dès le début à une plus grande précision. Nous procédons de la manière suivante :

Algorithme 5.3.2 : L'algorithme de Newton-Hensel stabilisé

Entrée : $n \geq 0$, $M < \frac{\rho}{(2^n+1)^2}$, $g_{\leq 2^n} + O(M, t^{2^n+1})$, $h_{\leq 2^n} + O(\frac{M}{2^n+1}, t^{2^n+1})$;

Sortie : u tel que $u' = g^{[0]}h^{[0]}(u) \mod (M2^n, t^{2^n})$.

début

| |
|--|
| $u \leftarrow t$; pour $i \in \llbracket 1, n \rrbracket$ faire <div style="border-left: 1px solid black; padding-left: 10px; margin-left: 10px;"> Remonter u à la précision $O(\frac{M}{2^i}, t^{2^{i-1}+1})$; $g_{< 2^{i-1}} \leftarrow (u'h(u)^{-1})_{< 2^{i-1}}$; $u \leftarrow u + h(u) \int \frac{1}{h(u)} (g(t)h(u) - u')$ mod t^{2^i+1} ; Retourner u ; </div> |
|--|

Correction : gestion des remontées

Le principal problème lorsqu'on effectue une méthode adaptative est de rester cohérent lorsque l'on augmente arbitrairement la précision. Plus précisément, dans l'Algorithme 5.3.2 nous augmentons la précision sur u jusqu'à $O(\frac{M}{2^i})$, et adaptons ensuite l'approximation sur g pour conserver la relation $g(t)h(u) - u' = 0 \pmod{(\frac{M}{2^i}, t^{2^{i-1}})}$.² Mais il reste la question de savoir comment ces g et u sont liés au $g^{[0]}$ originel et la solution recherchée y .

Pour cela, nous contrôlons la distance entre le $g^{[i]}$, valeur de g en entrant dans la boucle indexée par i , et le $g^{[0]}$ originel :

Lemme 5.3.3. *Après chaque mise à jour de g , nous avons $g^{[i]} = g^{[0]} + O(M)$, et $u^{[i]'} = g^{[i]}h(u^{[i]}) + O(M, t^{2^{i-1}})$.*

Démonstration. Nous montrons ce résultat par récurrence sur i , en ajoutant à l'hypothèse de récurrence que $u^{[i]} = u^{[0]} \pmod{t}$. Le résultat est clair pour $i = 0$. Supposons-le vrai pour un $i \in \mathbb{N}$.

Dans ce cas, au début de la boucle indexée par $i + 1$, $u^{[i]}$ est remonté à la précision $O(\frac{M}{2^i}, t^{2^{i-1}+1})$. Ensuite, $g^{[i+1]}$ est défini par $g_{\geq 2^{i-1}}^{[i+1]} = g_{\geq 2^{i-1}}^{[0]}$ et $g_{< 2^{i-1}}^{[i+1]} \leftarrow (u^{[i]'}h(u^{[i]})^{-1})_{< 2^{i-1}}$. Or, $u^{[i]'} = g^{[i]}h(u^{[i]}) + O(M, t^{2^{i-1}})$. Nous en déduisons que $g_{< 2^{i-1}}^{[i+1]} = g_{< 2^{i-1}}^{[i]} + O(M)$, et ainsi, nous avons bien $g^{[i+1]} = g^{[0]} + O(M)$.

Maintenant,

$$u^{[i+1]} = u^{[i]} + h(u^{[i]}) \int \frac{1}{h(u^{[i]})} \left(g^{[i+1]}(t)h(u^{[i+1]}) - u'^{[i+1]} \right) \pmod{t^{2^i+1}}.$$

Comme

$$g^{[i+1]}(t)h(u^{[i]}) - u'^{[i]} = O(\frac{M}{2^i}, t^{2^{i-1}}),$$

il est clair que $u^{[i+1]}$ est bien défini dans $\mathbb{Z}_p[[t]]$ et $u^{[i+1]} = u^{[0]} \pmod{t}$.

De plus, en développant comme dans la preuve du Lemme 5.3.1, nous obtenons directement que $u^{[i+1]'} = g^{[i+1]}h(u^{[i+1]}) + O(M, t^{2^i})$. \square

En sortie de l'algorithme, u est tel que $u' = g^{[0]}h(u) + O(M, t^{2^n})$.

D'après la Proposition 5.2.7, $u + O(M)$ est alors une approximation modulo $O(M2^n, t^{2^n})$ de la solution recherchée. Les chiffres p -adiques entre $M2^n$ et M ne sont pas nécessairement pertinents. Ceci achève la preuve du Théorème 5.1.2 hormis la partie complexité.

5.3.3. Complexité

Pour le dernier morceau restant dans la preuve du Théorème 5.1.2, nous étudions la complexité de l'Algorithme 5.3.2 La i -ème itération de l'algorithme met un jeu additions, multiplications, inversions et intégrations de séries formelles modulo (ε, t^{2^i}) . La complexité binaire de ces opérations est en $\mathcal{O}(M(2^i) |\log \varepsilon|)$. La i -ème itération de l'algorithme applique aussi une composition avec h , dont la complexité est par définition, $C_h(2^i, \varepsilon)$. Nous en déduisons que l'Algorithme 5.3.2 est bien en $\mathcal{O}(M(n) |\log \varepsilon| + C_h(\varepsilon, n))$. La preuve du Théorème 5.1.2 est ainsi achevée.

5.4. Implémentation

Nous avons implémenté cet algorithme dans le système de calcul formel SAGE [S⁺11]. Nous avons pu constater en pratique que l'Algorithme 5.3.2 atteint bien la perte de précision attendue.

Nous comparons dans le tableau suivant la précision requise pour le calcul d'isogénies dans [LS08] et celle requise par le Théorème 5.1.2. En entrée (l, p) du tableau correspond la précision requise pour le calcul de la solution de $y'^2 = gh(y)$ modulo t^{4l} dans $\mathbb{Z}_p[[t]]$, avec g, h tels qu'il existe une telle solution dans $\mathbb{Z}_p[[t]]$.

2. Ceci constitue un exemple de méthode adaptative où nous n'ajoutons pas seulement des chiffres zéros (ou arbitraires) sur toutes les entrées.

5. Un exemple complet : résolution d'équations différentielles

Notre borne sur la précision est présentée dans les colonnes nomées LV, sous la forme $x + y = z$ avec x la borne requise sur la précision en entrée et y la plus grande précision additionnelle à laquelle nous devons remonter durant l'itération de Newton stabilisée. La précision requise dans [LS08] est présentée dans les colonnes nommées LS.

| p | 5 | | 7 | | 11 | |
|-----|----|--------------|----|--------------|----|-------------|
| l | LS | LV | LS | LV | LS | LV |
| 67 | 14 | $5 + 3 = 8$ | 11 | $5 + 2 = 7$ | 9 | $5 + 2 = 7$ |
| 71 | 15 | $7 + 3 = 10$ | 11 | $5 + 3 = 8$ | 9 | $5 + 2 = 7$ |
| 89 | 15 | $7 + 3 = 10$ | 13 | $7 + 3 = 10$ | 9 | $5 + 2 = 7$ |
| 97 | 16 | $7 + 3 = 10$ | 13 | $7 + 3 = 10$ | 10 | $5 + 2 = 7$ |
| 131 | 22 | $7 + 4 = 11$ | 16 | $7 + 3 = 10$ | 12 | $5 + 2 = 7$ |
| 257 | 22 | $9 + 4 = 13$ | 16 | $7 + 3 = 10$ | 12 | $5 + 3 = 8$ |

Deuxième partie

Systèmes polynomiaux

Résumé

Soit $(f_1, \dots, f_s) \in \mathbb{Q}_p[X_1, \dots, X_n]^s$ des polynômes homogènes avec coefficients p -adiques. Un tel système peut apparaître, par exemple, en géométrie arithmétique. Cependant, comme \mathbb{Q}_p n'est pas un corps effectif, les algorithmes classiques ne peuvent s'appliquer directement. Dans ce contexte de précision finie, nous définissons ce que nous appelons une base de Gröbner approchée par rapport à un ordre monomial w .

Cette seconde partie est consacrée au calcul de telles bases de Gröbner approchées, selon trois stratégies pouvant interagir :

1. Un calcul direct par une adaptation de l'algorithme F5-Matriciel ;
2. Un calcul par changement d'ordre à partir d'une première base de Gröbner approchée et d'une adaptation de l'algorithme FGLM ;
3. Une approche tropicale des deux stratégies précédentes, où la valuation des coefficients est prise en compte, amenant une plus grande stabilité numérique.

Ces stratégies permettent le calcul de bases de Gröbner approchées dans de nombreux cas : génériquement ou conjecturalement génériquement, et une étude du comportement de la précision lors de ces calculs permet d'estimer quelle est la précision nécessaire pour pouvoir mener à bien le calcul d'une base de Gröbner approchée.

À cette fin, le Chapitre 6 constituera une courte introduction à la théorie des bases de Gröbner, ainsi qu'aux algorithmes maintenant classiques utilisés pour les calculer F5-Matriciel et FGLM. Le Chapitre 7 présentera une analyse directe de l'algorithme F5-Matriciel dans le contexte des corps complets discrètement valués à précision finie, ainsi que quelques applications de cette analyse à des calculs sur les rationnels. Le Chapitre 8 est consacré à l'étude de l'algorithme FGLM dans ce même contexte. Enfin, le Chapitre 9 apporte une approche tropicale à l'étude des bases de Gröbner, l'appliquant aux algorithmes F5-Matriciel et FGLM. Il permettra de définir une stratégie passant par un calcul de base de Gröbner tropicale pour obtenir une base de Gröbner classique, lorsque les composantes homogènes de plus haut degré des polynômes en entrée forment une suite régulière.

Contexte

La question qui structure cette partie est celle-ci : quels calculs de bases de Gröbner sont possibles sur \mathbb{Q}_p ? Des questions proches ont déjà été posées et, au moins partiellement, résolues.

Méthodes p -adiques et bases de Gröbner

Historiquement, la première d'entre elles est la suivante : est-ce que passer par \mathbb{Q}_p lors d'un calcul de bases de Gröbner sur \mathbb{Q} peut accélérer ce calcul ?

L'idée est alors initialement d'appliquer une méthode p -adique classique pour le calcul d'une base de Gröbner G pour un ordre monomial w d'un système polynomial F à coefficients entiers :

1. Trouver un nombre premier p adapté au calcul, soit essentiellement tel que $LM(\langle F \rangle) = LM(\langle \bar{F} \rangle)$ où \bar{F} est la réduction modulo p de F .
2. Calculer une base de Gröbner \bar{G} de \bar{F} pour w , en même temps qu'une matrice \bar{A} telle que $\bar{G} = \bar{A} \cdot \bar{F}$.
3. Remonter l'égalité $\bar{G} = \bar{A} \cdot \bar{F}$ dans $\mathbb{Z}/p^n\mathbb{Z}$.
4. Lorsque n est assez grand, en déduire une base de Gröbner G de $\langle F \rangle$, éventuellement par reconstruction rationnelle des coefficients.

Cette approche rencontre deux écueils majeurs. Le premier est le choix de p , qui doit vérifier la condition $LM(\langle F \rangle) = LM(\langle \bar{F} \rangle)$. La terminologie classique est celle d'un p "chanceux", et elle n'est pas usurpée puisqu'il n'y a pas *a priori* de moyen de savoir si cette condition est remplie. Heureusement, pour un F donné, il n'est tout de même pas difficile de voir que seul un nombre fini de p ne sont pas chanceux.

Le second écueil est de savoir quand est-ce que n est assez grand pour l'on puisse bien reconnaître les coefficients d'un G qui convient. Il n'y a en effet là encore pas d'estimation *a priori* suffisante.

Néanmoins, ces écueils ne sont pas nécessairement rédhibitoires. En effet, il est souvent aisé (voir presque gratuit) de vérifier *a posteriori* que p est chanceux ou que la précision est suffisante. Posséder une bonne estimation *a priori* de cette précision permet d'optimiser le temps de calcul, mais tester une borne, même éventuellement grossière, et tester *a posteriori* peut dans bien des cas donner des résultats bien suffisants.

Nous renvoyons aux travaux de Winkler [Win88], Pauer [Pau92], Gräbe [Grä93], Arnold [Arn03], et Renault et Yokoyama [RY06] pour plus de détails et des applications de ces méthodes. Nous remarquons cependant qu'aucun de ces travaux ne s'intéresse aux calculs de bases de Gröbner sur \mathbb{Q}_p pour elles-mêmes, mais seulement comme moyen vers des calculs sur \mathbb{Q} .

Bases de Gröbner flottantes

Une deuxième question, liée à la notre et qui se pose naturellement est celle de savoir quels calculs de bases de Gröbner sont possibles sur les flottants. En effet, les calculs polynomiaux apparaissent dans diverses applications concrètes utilisant les flottants, et pour des raisons de géométrie ou de résolutions de systèmes, vouloir calculer des bases de Gröbner sur les flottants paraît intéressant.

Les flottants étant, de même que les p -adiques, des objets ne pouvant être manipulés qu'à précision finie, ce problème n'est pas évident. En effet, il est essentiel dans la plupart des calculs de bases de Gröbner de réduire des polynômes ou des coefficients à zéro, ce qui est difficile, voire impossible sur les flottants.

Néanmoins, de nombreux auteurs ont contribué à l'étude de ce problème, notamment Shirayanagi & Sweedler [SS98], Kondratyev, Stetter & Winkler [KSW04], Nagasaka [Nag09], Stetter [Ste], Traverso & Zanzi [TZ02], et bien d'autres. Afin de palier au problème de la précision, diverses techniques y sont proposées, notamment l'ajout d'une nouvelle variable ε (et le fait de travailler dans $\mathbb{R}[\varepsilon]/\varepsilon^2$). Une très bonne introduction au domaine du calcul des bases de Gröbner numériques est fournie par Sasaki et Kako dans [SK07], [SK10] et Sasaki dans [Sas11]. Y est notamment introduit une classification des compensations (*cancellation*) à précision finie qui peuvent apparaître lors du calcul. Certaines sont bien des approximations de vrais zéros, tandis que d'autres sont accidentelles, et ne sont dues qu'au manque de précision, qui ne permet pas de distinguer qu'un coefficient est non nul. Diverses méthodes sont proposées pour réduire la quantité de compensation accidentelles, notamment une réduction par algorithme de Gauss de l'écriture des polynômes considérés en fonction des polynômes en entrée, afin, en choisissant ses pivots, de minimiser les pertes de précision.

Remarquons cependant que ces travaux prennent toujours pour point de vue celui des flottants, dont le comportement vis-à-vis de la précision n'est pas identique à celui de corps comme \mathbb{Q}_p ou des séries formelles.

Néanmoins, une méthode appelée TSV (*Term Substitutions with Variables*, remplacement de termes par des variables) et qui est *a priori* disponible sur tous corps a été développée par Faugère et Liang dans [FL07, FL11a, FL11b]. Le principe est le suivant : si après réduction, le coefficient de tête d'un polynôme lors du calcul d'une base de Gröbner pose un problème de précision, par exemple pour $\varepsilon x^3 + 2y^3$ pour l'ordre grevlex avec $x > y$ sur $\mathbb{Q}[x, y]$, le monôme de tête est remplacé par une nouvelle variable et un polynôme est ajouté pour tenir compte de la substitution.³ Dans l'exemple considéré, $\varepsilon x^3 + 2y^3$ est remplacé par $z + 2y^3$ et $x^3 - z$, et l'ordre monomial utilisé est alors l'ordre grevlex avec $x > y > z$ sur $\mathbb{Q}[x, y]$. Pour un idéal de dimension zéro, s'il est possible d'obtenir une base de Gröbner sans ambiguïté après un nombre fini de tels ajouts de variables, alors, essentiellement, les solutions peuvent être lues sur celles de cette base de Gröbner. Remarquons que cette méthode est efficace pour trouver les solutions d'un système, mais ne permet pas d'obtenir une base de Gröbner de l'idéal de départ.

Résolution de systèmes polynomiaux p -adiques

Enfin, une dernière question liée à celle qui nous intéresse est de savoir quels systèmes polynomiaux peuvent être résolus en p -adiques.

L'école de l'algorithmique p -adique détendue s'est intéressée à cette question, notamment avec Berthomieu et Lebreton dans [BL12] et Lebreton dans [Leb13]. Dans ces articles, modulo essentiellement une hypothèse de bonne réduction, les auteurs montrent qu'il est possible de remonter

3. En faisant cela, on ajoute un nouveau générateur à la liste courante des générateurs.

une racine modulo p ou une représentation univariée d'un idéal *via* le formalisme des p -adiques récursifs et de l'algorithmique détendue. Ceci est particulièrement intéressant si, étant donné une représentation rationnelle d'un idéal sur les rationnels, il nous est laissé le choix d'un p donnant une bonne réduction de la représentation. De cette manière, les racines rationnelles peuvent être efficacement obtenue par ces méthodes p -adiques. Cependant, celles-ci ne s'appliquent pas forcément bien pour obtenir des racines p -adiques qui ne se réduiraient pas bien modulo p .

D'autres méthodes de résolution de systèmes polynomiaux existent : méthodes dites de *réécritures* comme les bases de bord (voir [MT08]), ou dites *d'évaluations* comme les représentations de Kronecker (voir [GLS01], [DS04] et [DL08]) ou univariées rationnelles (voir [Rou99]), et il serait intéressant d'étudier leur comportement en p -adique à précision finie, ce que nous ne ferons pas ici.

Notations

Nous présentons ici les notations principales utilisés dans cette partie. Afin de ne pas entraîner de confusions entre les propositions portant sur des objets de nature théorique et ceux sur lesquels nous étudions la précision en pratique, nous utiliserons des notations distinctes.

Notations en précision infinie

Dans tout ce qui suit, \mathcal{K} est un corps, $n \in \mathbb{N}^*$ et $\mathcal{A} = \mathcal{K}[X_1, \dots, X_n]$. Nous noterons \mathcal{A}_d l'ensemble des polynômes homogènes de degré d de \mathcal{A} , $\mathcal{A}_{\leq d}$ pour les polynômes de \mathcal{A} de degré total inférieur ou égal à d , et $\mathcal{A}_{\geq d} = \bigoplus_{k \geq d} \mathcal{A}_k$. Si $u = (u_1, \dots, u_n) \in \mathbb{Z}_{\geq 0}^n$, nous écrirons X^u pour $X_1^{u_1} \dots X_n^{u_n}$. Si $P \in \mathcal{A}$ est un polynôme homogène, nous notons $|P|$ pour son degré. En-dehors du Chapitre 6, \mathcal{K} est muni d'une valuation discrète val qui le rend complet.

Notations en précision finie

Dans tout ce qui suit, K est un corps, $n \in \mathbb{N}^*$ et $A = K[X_1, \dots, X_n]$. De plus, nous supposons que K est un corps discrètement valué pour une valuation val . Nous demandons que K soit complet par rapport à la norme définie par val . Soit $\Gamma = val(K \setminus \{0\})$. Nous notons $R = O_K$ l'anneau des entiers de K , m_K son idéal maximal et $k_K = O_K/m_K$ son corps résiduel. Nous noterons CDVF (*complete discrete-valuation field*) un tel corps. Nous renvoyons à Corps Locaux, de Serre [Ser62], pour une introduction à de tels corps. Soit $\pi \in R$ une uniformisante de K et soit $S_K \subset R$ un système de représentants de $k_K = O_K/m_K$, avec $0 \in S_K$. Tout élément de K peut s'écrire de manière unique suivant un développement π -adique de la forme $\sum_{k \geq l} a_k \pi^k$, avec $l \in \mathbb{Z}$, et des $a_k \in S_K$.

Plus particulièrement, le cas qui nous intéresse est celui où K n'est pas nécessairement un corps effectif, mais où son corps résiduel k_K l'est (*i.e.* nous disposons d'algorithmes pour toutes les opérations de corps et pour tester l'égalité entre deux éléments de k). Ainsi, des calculs formels, ou à précision infinie, peuvent être effectués sur des troncatures de développement π -adiques d'éléments de K . Nous appelons CDVF à précision finie un tel corps, et CDVR à précision finie pour son anneau des entiers. Des exemples classiques de CDVF à précision finie sont $K = \mathbb{Q}_p$, avec valuation p -adique, et $\mathbb{Q}[[X]]$ ou $\mathbb{F}_q[[X]]$ avec valuation X -adique. Nous supposons dorénavant que K est un CDVF à précision finie.

De plus, avec $A = K[X_1, \dots, X_n]$, nous écrivons aussi $B = R[X_1, \dots, X_n]$. Nous notons A_d pour l'ensemble des polynômes homogènes de degré d de A , $A_{\leq d} = \bigoplus_{k=0}^d A_k$, et $A_{\geq d} = \bigoplus_{k \geq d} A_k$. Si $u = (u_1, \dots, u_n) \in \mathbb{Z}_{\geq 0}^n$, nous écrirons X^u pour $X_1^{u_1} \dots X_n^{u_n}$. Si $P \in A$ est un polynôme homogène, nous notons $|P|$ pour son degré.

Modèle de complexité

Nous renvoyons à la discussion de la partie précédente, à la page 32, et adoptons le même modèle.

Synthèse des résultats de cette partie

Nous présentons ici plusieurs tableaux résumant les résultats obtenus dans cette partie, ainsi que quelques schémas illustrant comment les différents algorithmes peuvent être composés pour calculer des bases de Gröbner.

Calcul direct d'une base de Gröbner

Les Chapitres 7 et 9 s'intéressent au calcul direct, respectivement d'une base de Gröbner et d'une base de Gröbner tropicale. Nous présentons dans le tableau suivant les résultats du Théorème 7.1.1 et de la Proposition 9.1.2, qui énoncent que sur des ouverts de Zariski (donnés par la condition sur les polynômes en entrée donnée dans la ligne *hypothèse*), il est possible de calculer des bases de Gröbner approchées. Il est précisé (sur les lignes *précision* puis *remarque*) comment sont estimées les pertes de précision, ainsi que le temps de calcul, en nombre d'opérations sur le corps, à la précision initiale.

| | F5-Matriciel Faible | F5-Matriciel Tropical | F5-Matriciel Tropical, $w = 0$ |
|-----------------|--------------------------------------|--------------------------------------|--------------------------------------|
| hypothèse | H1 et H2 | H1 | H1 |
| précision | $prec_{F5M}(F, \omega)$ | $prec_{F5Mtrop}(F, \leq, w)$ | $prec_{F5Mtrop}(F, \leq, 0)$ |
| remarque | mineurs des matrices de Macaulay | | plus petite valuation des mineurs |
| temps de calcul | $O\left(sD\binom{n+D-1}{D}^3\right)$ | $O\left(sD\binom{n+D-1}{D}^3\right)$ | $O\left(sD\binom{n+D-1}{D}^3\right)$ |

Ici, **H1** correspond à une hypothèse de régularité et **H2** de généricité, qui est pour l'instant seulement conjecturalement générique pour l'ordre grevlex. Dans ce tableau, s est le nombre de polynômes initiaux, n le nombre de variables de l'anneau de polynômes dans lequel ils vivent et le calcul effectué est celui de D -bases de Gröbner. De plus ω et \leq sont des ordres monomiaux et w un poids pour un ordre tropical.

Réduction d'une base de Gröbner

Pour ce qui est de calculer une base de Gröbner réduite à partir d'une base de Gröbner G (pour un ordre monomial ω), il n'y a pas d'hypothèse autre que sur la précision des polynômes en entrée. Ce résultat est obtenu au Théorème 7.6.7. La notion de conditionnement ici, $cond(G, \omega)$ se lit là encore sur les pivots sur la matrice de Macaulay déduite de G .

| | GB vers GB réduite |
|-----------------|----------------------------------|
| hypothèse | \emptyset |
| précision | $cond(G, \omega)$ |
| remarque | somme des val des LC par degré |
| temps de calcul | (négligeable devant les autres) |

Changement d'ordre en dimension zéro

Une fois qu'une base de Gröbner approchée est calculée pour un ordre donné, et si l'idéal correspondant est de dimension zéro, alors le Théorème 8.1.6 nous montre que nous pouvons en déduire une base de Gröbner pour n'importe quel autre ordre monomial, dès que la précision est suffisante. Il est raffiné avec le Théorème 8.2.4 pour les cas particuliers des ordres lexicographiques et grevlex. Le premier ordre peut en fait être un ordre tropical sur les termes, avec la Proposition 9.3.13.

| | FGLM | FGLM vers lex | FGLM grevlex vers lex |
|-----------------|--|--------------------|--------------------------------|
| hypothèse | \emptyset | position générale | généricité + position générale |
| précision | $cond_{\leq, \leq_2}$ | $cond_{\leq, lex}$ | $cond_{grevlex, lex}$ |
| remarque | max des val des facteurs invariants de la matrice de passage | | |
| temps de calcul | $O(n\delta^3)$ | | $O(\delta^3) + O(n\delta^2)$ |

Ici, n est le nombre de variables de l'anneau de polynômes A concerné, et δ est le degré de l'idéal I étudié. Les notions de conditionnement (ligne *précision*) sont données, à chaque fois, par la plus

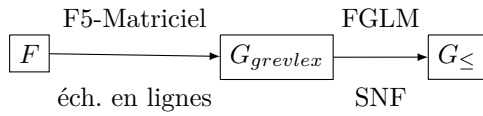
grande valuation d'un facteur invariant pour la matrice de changement de base entre les bases canoniques de A/I pour les deux ordres considérés.

Schémas de calcul d'une base de Gröbner en dimension zéro

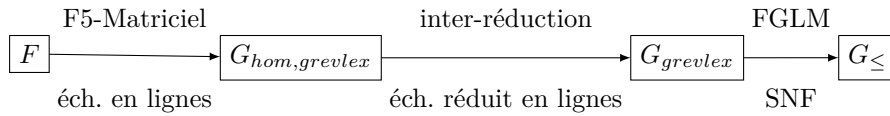
Nous pouvons maintenant fournir une vision globale et une comparaison des méthodes développées dans cette thèse pour le calcul de bases de Gröbner sur des CDVF à précision finie. Remarquons que, en un sens que nous précisons, les hypothèses énoncées forment un ouvert sur lequel le calcul de base de Gröbner (par les schémas proposés) est continu.

Calcul direct d'une base de Gröbner

Pour le calcul d'une base de Gröbner pour un ordre \leq de l'idéal engendré par une famille de polynômes homogènes F satisfaisant les hypothèses **H1** et **H2** (pour grevlex), et engendrant un idéal de dimension zéro, le schéma est le suivant :



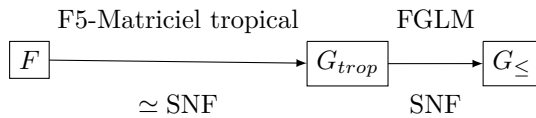
Si F n'est pas constitué de polynômes homogènes, mais que les composantes homogènes de plus haut degré de ceux-ci vérifient les hypothèses **H1** et **H2** (pour grevlex), le schéma est le suivant :



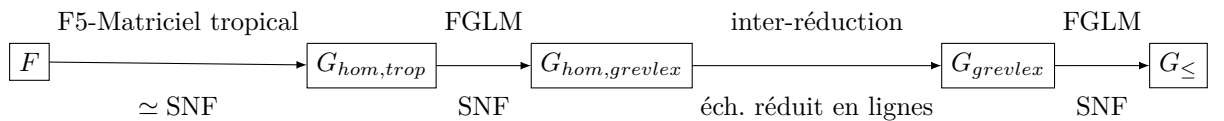
où $G_{\text{hom,grevlex}}$ est une base de Gröbner de l'idéal engendré par les composantes homogènes de plus haut degré de F .

Calcul d'une base de Gröbner par une méthode tropicale

Si l'on souhaite calculer une base de Gröbner pour un ordre \leq de l'idéal engendré par une famille de polynômes homogènes F satisfaisant l'hypothèses **H1** et engendrant un idéal de dimension zéro, on choisit l'ordre sur les termes engendré par $w = (0, \dots, 0)$ et un ordre *grevlex*, et le schéma est alors le suivant :



Finalement, si F n'est pas constitué de polynômes homogènes, mais que les composantes homogènes de plus haut degré de ceux-ci vérifient l'hypothèse **H1** et que F engendre un idéal de dimension zéro, le schéma est le suivant :



Bases de Gröbner tropicales en précision infinie

Enfin, pour le cas particulier où l'on travaille à précision infinie et que l'on cherche à calculer une base de Gröbner tropicale d'une suite régulière homogène, grâce à la Proposition 9.1.3, un algorithme F5-Matriciel Tropical avec signature est disponible, atteignant une complexité comparable à celle pour le cas classique :

| | |
|-----------------|--------------------------------------|
| | F5-Matriciel Tropical avec signature |
| hypothèse | précision infinie |
| précision | \emptyset |
| remarque | \emptyset |
| temps de calcul | $O\left(D\binom{n+D-1}{D}^3\right)$ |

Ici encore, n est le nombre de variables et D est le degré de troncature (on calcule des D -bases de Gröbner).

6. Algorithmes classiques pour le calcul de bases de Gröbner

"It's dangerous to go alone! Take this."

Old man, *The Legend of Zelda*

Le but de ce chapitre est de présenter une courte introduction à la théorie des bases de Gröbner¹, ainsi qu'aux deux algorithmes classiques, F5-Matriciel et FGLM, dont nous étudierons par la suite la stabilité à précision finie. La Section 6.1 s'attache ainsi à fournir une présentation de ce que sont les bases de Gröbner, ainsi que l'intérêt de leurs calculs. Ensuite, la Section 6.2 présente les algorithmes de calcul de bases de Gröbner F5-Matriciel et FGLM.

6.1. Bases de Gröbner et applications

6.1.1. Que sont les bases de Gröbner ?

Les bases de Gröbner sont un outil d'algèbre commutative développé par Bruno Buchberger (voir [Buc65]) dans les années 1960². Elles permettent de résoudre algorithmiquement la plupart des questions et calculs que l'on peut vouloir faire sur des idéaux dans des anneaux de polynômes. Elles permettent aussi une généralisation multivariée de la division euclidienne sur un anneau de polynômes en une variable.

Leur important développement ces dernières décennies concorde avec celui du calcul formel, et elles sont maintenant présentes tant théoriquement qu'effectivement dans des domaines aussi variés que la géométrie algébrique, la cryptographie, les équations aux dérivées partielles ou l'informatique.

Ordres monomiaux, division et non-euclidianité

Pour pouvoir définir ce que sont les bases de Gröbner, la première étape est de trouver un moyen raisonnable d'ordonner totalement les monômes dans un anneau de polynômes.

Définition 6.1.1. Nous appelons ordre monomial de \mathcal{A} un ordre total \leq sur les monômes de \mathcal{A} , vérifiant les propriétés suivantes :

1. pour tous monômes $x^\alpha, x^\beta, x^\gamma$, si $x^\alpha \leq x^\beta$ alors $x^\alpha x^\gamma \leq x^\beta x^\gamma$;
2. Tout ensemble non-vide de monômes admet un plus petit élément.

Deux ordres monomiaux classiques sur \mathcal{A} sont l'ordre lexicographique, noté lex , et l'ordre lexicographique inverse gradué, noté grevlex , dont voici les définitions :

Définition 6.1.2. L'ordre lexicographique sur \mathcal{A} pour $X_1 > \dots > X_n$, noté \leq_{lex} , est défini par : si x^α et x^β sont deux monômes, alors $x^\alpha \leq_{\text{lex}} x^\beta$ lorsque le premier coefficient non nul, en partant de la gauche, de $\alpha - \beta$ est négatif.

Définition 6.1.3. L'ordre grevlex sur \mathcal{A} pour $X_1 > \dots > X_n$, noté \leq_{grevlex} , est défini par : si x^α et x^β sont deux monômes, alors $x^\alpha \leq_{\text{grevlex}} x^\beta$ lorsque $|x^\alpha| < |x^\beta|$, ou $|x^\alpha| = |x^\beta|$ et le premier coefficient non nul, en partant de la droite, de $\alpha - \beta$ est positif.

1. Nous renvoyons à [CLO07] pour une introduction bien plus détaillée à cette théorie.

2. Heisuke Hironaka a aussi travaillé de manière indépendante sur des questions similaires à la même époque. Le mathématicien géorgien N.M.Gjunter a aussi introduit une notion similaire en 1913. Néanmoins, parce qu'il a fourni le premier algorithme effectif de calcul de ces bases, Buchberger en conserve la paternité dans la littérature.

L'ordre grevlex est un ordre monomial qui raffine le degré, au sens suivant :

Définition 6.1.4. Soit \leq un ordre monomial sur \mathcal{A} . On dit que \leq raffine le degré si, pour x^α et x^β deux monômes de \mathcal{A} , $|x^\alpha| < |x^\beta|$ implique $x^\alpha \leq x^\beta$.

Comme conséquence du fait qu'un ordre monomial \leq est total, nous avons que tout ensemble fini de monômes admet un plus grand élément. Ceci permet de définir les notions de terme de tête d'un polynôme :

Définition 6.1.5. Soit $f = \sum_{\alpha} c_{\alpha} x^{\alpha} \in \mathcal{A}$ et \leq un ordre monomial sur \mathcal{A} . Alors, nous définissons le monôme de tête, coefficient de tête et terme de tête comme suit :

- $LM_{\leq}(f) = \max_{\leq} \{x^{\alpha} | c_{\alpha} \neq 0\}$;
- $LC_{\leq}(f) = c_{\alpha}$ pour le α tel que $x^{\alpha} = LM_{\leq}(f)$;
- $LT_{\leq}(f) = LC_{\leq}(f) LM_{\leq}(f)$.

Exemple 6.1.6. Dans $\mathbb{Q}[x, y, z]$, pour l'ordre lexicographique, $LM(x^2 + x^2yz + y^4) = x^2yz$, tandis que pour l'ordre grevlex, $LM(x^2 + x^2yz + y^4) = y^4$

Étant donné un idéal, nous pouvons regarder les termes de tête de ses polynômes de la façon suivante :

Définition 6.1.7. L'idéal de tête d'un idéal I , noté $LM_{\leq}(I)$ est défini par :

$$LM_{\leq}(I) = \langle \{LM_{\leq}(f) | f \in I\} \rangle.$$

Remarque 6.1.8. Nous noterons parfois $LM(f)$, $LC(f)$, $LT(f)$ et $LM(I)$ lorsqu'il n'y a pas d'ambiguïté sur le choix de l'ordre monomial.

Division dans un anneau de polynômes

Les définitions qui précèdent nous permettent de définir l'algorithme suivant, algorithme de division d'un polynôme par une famille de polynômes, qui généralise au cas multivarié l'algorithme de division euclidienne.

Algorithme 6.1.9 : Division d'un polynôme par une famille de polynômes

entrée : $f \in \mathcal{A}$, $F = (f_1, \dots, f_s) \in \mathcal{A}^s$ des polynômes.

sortie : $r \in \mathcal{A}$, $(q_1, \dots, q_s) \in \mathcal{A}^s$ tels que $f = r + \sum_{i=1}^s q_i f_i$ et aucun monôme de r n'est divisible par un des $LM(f_i)$.

début

```

     $h := f$  ;
     $i := 1$  ;
     $(q_1, \dots, q_s) := (0, \dots, 0)$  ;
    tant que  $h \neq 0$  faire
        si  $i \leq s$  alors
            si  $LM(f_i)$  divise  $LM(h)$  alors
                 $q_i := q_i + \frac{LT(h)}{LT(f_i)}$  ;
                 $h := h - \frac{LT(h)}{LT(f_i)} f_i$  ;
                 $i := 1$  ;
            sinon
                 $i := i + 1$ 
            sinon
                 $r := r + LT(h)$  ;
                 $h := h - LT(h)$  ;
                 $i := 1$ 
        Retourner  $r, (q_1, \dots, q_s)$  ;

```

Définition 6.1.10. Nous appelons r le reste dans la division de f par (f_1, \dots, f_s) et (q_1, \dots, q_s) le quotient. Il nous faut remarquer que l'ordre des f_i ainsi que le choix de l'ordre monomial influe sur les valeurs de r et de (q_1, \dots, q_s) . r pourra être noté $f \bmod (f_1, \dots, f_s)$.

Exemple 6.1.11. Dans $\mathbb{Q}[x, y]$ muni de l'ordre lexicographique donné par $x > y$, pour $f = xy^2 - x$ et $(f_1, f_2) = (xy + 1, y^2 - 1)$, nous avons $f \bmod (f_1, f_2) = -x - y$ et $f \bmod (f_2, f_1) = 0$.

\mathcal{A} n'est un anneau euclidien que si $n = 1$, et on ne peut espérer que cet algorithme de division dans un cas multivarié possède d'aussi bonnes propriétés que celui de division euclidienne. Néanmoins, lorsqu'on divise par une famille de polynôme qui se trouve être une base de Gröbner, nous arrivons à en retrouver certaines.

Bases de Gröbner

Les définitions précédentes sont suffisantes pour pouvoir définir la notion de base de Gröbner d'un idéal :

Définition 6.1.12. Soit I un idéal de \mathcal{A} et \leq un ordre monomial sur \mathcal{A} . Une base de Gröbner de I pour \leq est un sous-ensemble $G \subset I$ tel que $LM_{\leq}(I) = \langle LM(g) | g \in G \rangle$.

L'algorithme de division permet alors de résoudre le problème de l'appartenance à un idéal :

Proposition 6.1.13. Soit $f \in \mathcal{A}$ et $G = (g_1, \dots, g_t) \in \mathcal{A}^t$ des polynômes tels que G forme une base de Gröbner de $I = \langle g_1, \dots, g_t \rangle$. Alors $f \in I$ si et seulement si $f \bmod G = 0$.

Parmi les bases de Gröbner, certains sont particulières :

Définition 6.1.14. Soit G une base de Gröbner de l'idéal I pour un ordre monomial \leq . Alors G est dite minimale si pour tout couple (g, h) dans G , $LM(g)$ divise $LM(h)$ si et seulement si $g = h$.

Définition 6.1.15. Soit G une base de Gröbner de l'idéal I pour un ordre monomial \leq . Alors G est dite réduite si, d'une part, pour tout couple (g, h) dans G , avec $g \neq h$, pour tout x^α monôme de g , $LM(h)$ ne divise pas x^α , et d'autre part, pour tout $g \in G$, $LC(g) = 1$.

Nous avons alors la propriété d'unicité suivante :

Proposition 6.1.16. Soit I un idéal de \mathcal{A} et \leq un ordre monomial sur \mathcal{A} . Alors I possède, à permutation de ses éléments près, une et une seule base de Gröbner réduite pour \leq .

Pour ce qui est du problème du calcul d'une base de Gröbner d'un idéal, Buchberger a, en premier, proposé dans [Buc65] un algorithme le résolvant. Il s'agit du bien connu algorithme de Buchberger, que nous ne rappellerons pas ici car il ne nous sera pas utile. Nous renvoyons néanmoins à [CLO07] pour plus de détails. Pour toute famille finie de polynômes F , l'algorithme de Buchberger calcule une base de Gröbner de l'idéal $\langle F \rangle$, mais cependant, l'étude de cet algorithme, tant en ce qui concerne sa complexité que sa stabilité, n'est pas aisée, et c'est pourquoi nous n'en parlerons pas plus. La Section 6.2 est consacrée à la présentation de quelques algorithmes modernes de calcul de bases de Gröbner.

6.1.2. Pourquoi calculer des bases de Gröbner ?

Obtenir une base de Gröbner d'un idéal permet de résoudre de nombreux problèmes. Nous avons choisi d'en présenter quelques-uns dans ce qui suit.

Calculs dans le quotient

Une première application, peut-être l'une des plus naturelles, est que le calcul de bases de Gröbner permet de travailler dans le quotient par un idéal :

Proposition 6.1.17. Soit $I \subset \mathcal{A}$, \leq un ordre monomial, et G une base de Gröbner de I pour \leq . Alors :

$$\begin{aligned} \mathcal{A} &\rightarrow \mathcal{A}/I \\ f &\mapsto f \bmod G \end{aligned}$$

est un morphisme surjectif d'anneaux, de noyau I .

6. Algorithmes classiques pour le calcul de bases de Gröbner

En complément de la proposition suivante, lorsque I est de dimension zéro, $B_{\leq} = \{x^\alpha | x^\alpha \notin \langle LM(G) \rangle\}$ forme une \mathcal{K} -base de \mathcal{A}/I , appelée base canonique de \mathcal{A}/I pour \leq . Par définition, la dimension de \mathcal{A}/I est le degré de I , et est aussi le cardinal de B_{\leq} . Elle peut donc être déduite de la connaissance de G .

Élimination

L'une des applications les plus importantes des bases de Gröbner est qu'elles permettent de réaliser l'élimination de variables.

Définition 6.1.18. Soit I un idéal de \mathcal{A} et soit $U \subset \{x_1, \dots, x_n\}$ une partie des variables de \mathcal{A} . Alors l'idéal d'élimination de I par rapport à U , noté I_U est $I \cap \mathcal{K}[U]$.

Un calcul de base de Gröbner permet de calculer les idéaux d'élimination. Pour cela, nous avons besoin de la notion d'ordre d'élimination.

Définition 6.1.19. Soit $U \subset \{x_1, \dots, x_n\}$ une partie des variables de \mathcal{A} . Alors, \leq est un ordre d'élimination pour U lorsque si $x^\alpha \in \mathcal{K}[U]$ et $x^\beta \in \mathcal{A} \setminus \mathcal{K}[U]$ sont deux monômes, alors $x^\alpha < x^\beta$.

Nous avons alors la proposition suivante pour calculer les idéaux d'élimination :

Proposition 6.1.20. Soit I un idéal de \mathcal{A} et soit $U \subset \{x_1, \dots, x_n\}$ une partie des variables de \mathcal{A} . Soit \leq un ordre d'élimination pour U et soit G une base de Gröbner de I pour \leq . Alors $G \cap \mathcal{K}[U]$ est une base de Gröbner de $I_U = I \cap \mathcal{K}[U]$.

L'ordre lexicographique fournit un exemple d'ordre d'élimination pour $U = \{x_n\}$ ou même $U = \{x_k, x_{k+1}, \dots, x_n\}$ pour tout $k \in \llbracket 1, n \rrbracket$.

Exemple 6.1.21. Soit $F = (x^2 + y + z - 1, x + y^2 + z - 1, x + y + z^2 - 1) \in \mathbb{Q}[x, y, z]$. Alors on peut montrer que $G = (x + y + z^2 - 1, y^2 - y - z^2 + z, 2yz^2 + z^4 - z^2, z^6 - 4z^4 + 4z^3 - z^2)$ est une base de Gröbner pour l'ordre lexicographique (avec $x > y > z$) de l'idéal $I = \langle F \rangle$. En conséquence de la proposition précédente, $(z^6 - 4z^4 + 4z^3 - z^2)$ est une base de Gröbner de l'idéal $I \cap \mathbb{Q}[z]$. Ceci permet de calculer toutes les solutions au système donné par F . Elles se déduisent toutes du cas univarié et des racines de $z^6 - 4z^4 + 4z^3 - z^2$.

Résoudre un système

Une autre application parmi les plus importantes des bases de Gröbner est celle de la résolution de systèmes polynomiaux. Ceci systématise, en un certain sens, ce qui est développé au paragraphe précédent. Par exemple, nous verrons dans la Sous-Sous-Section 6.2.2 que beaucoup d'idéaux de dimension zéro (ceux dit en position générale) admettent une base de Gröbner pour l'ordre lexicographique qui est de la forme suivante : $(x_1 - h_1(x_n), \dots, x_{n+1} - h_{n+1}(x_{n-1}), h_n(x_n))$, avec $\deg(h_n) = \delta$ et $\deg(h_i) < \delta$ pour $i \in \llbracket 1, n-1 \rrbracket$. Ceci ramène la résolution d'un système générant un idéal en position générale à celle d'un seul polynôme univarié, h_n .

Et bien d'autres choses ...

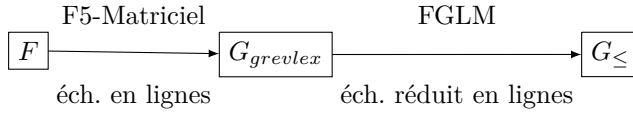
Les bases de Gröbner ont de nombreuses autres applications, comme le calcul de la dimension d'un idéal ou le calcul d'opérations sur les idéaux. Nous en donnons un dernier exemple avec le calcul de l'intersection :

Proposition 6.1.22. Soit I et J deux idéaux de \mathcal{A} . Soit G une base de Gröbner de l'idéal $tI + (1-t)J$ dans $\mathcal{K}[X_1, \dots, X_n, t]$ pour un ordre d'élimination pour $\{X_1, \dots, X_n\}$. Alors $G \cap \mathcal{A}$ est une base de Gröbner de $I \cap J$.

3. On peut les injecter dans les équations uniquement en y et z pour déterminer les y possibles (il y a alors une nouvelle équation univariée à résoudre), puis dans la première pour conclure.

6.2. Calcul de bases de Gröbner et algèbre linéaire

Maintenant que nous avons présenté ce que sont les bases de Gröbner et leurs applications, il nous reste à expliquer comment, étant donnée une famille de polynôme F , on peut calculer une base de Gröbner de l'idéal engendré par F . Comme souvent en mathématiques, une méthode efficace pour traiter un problème est de le ramener à une question d'algèbre linéaire, domaine plutôt bien maîtrisé. Dans ce qui suit, nous verrons qu'il y a deux moyens naturels de ramener le calcul de bases de Gröbner à un problème d'algèbre linéaire : travailler dans les gradués de \mathcal{A} (les \mathcal{A}_d) ou travailler dans l'anneau quotient, vu comme espace vectoriel. La première idée amène à l'algorithme F5-Matriciel tandis que la seconde amène à l'algorithme FGLM.⁴ Un schéma classique de calcul des bases de Gröbner, au moins dans le cas où F est homogène et engendre un idéal de dimension zéro, est alors le suivant :



avec G_{grevlex} une base de Gröbner de $\langle F \rangle$ pour un ordre *grevlex* et G_{\leq} une correspondant à l'ordre souhaité.

6.2.1. L'algorithme F5-Matriciel

Dans cette Sous-Section, nous exposons comment, avec les algorithmes de Lazard et F5-Matriciel, il est possible de ramener le calcul de bases de Gröbner à un simple échelonnement de matrices. Ceci permettra une analyse simple de la complexité, et plus tard, de la stabilité ou de la continuité du calcul de bases de Gröbner.

Algorithme de Lazard

La première idée pour aller vers l'algorithme F5-Matriciel est due à Daniel Lazard. Elle peut s'exprimer avec la remarque suivante : si $I \subset \mathcal{A}$ est un idéal homogène, engendré par des polynômes homogènes (f_1, \dots, f_s) , alors, pour tout $d \in \mathbb{N}$, on a l'égalité de \mathcal{K} -espaces vectoriels $I \cap \mathcal{A}_d = \langle x^\alpha f_i, |\alpha| + |f_i| = d \rangle$. Cette égalité amène l'implication suivante : degré par degré, on peut exprimer nos calculs en terme de matrices. Ainsi, nous allons pouvoir ramener notre problème à une étude matricielle. Pour cela, nous définissons les matrices de Macaulay :

Définition 6.2.1. Soit $B_{n,d} = (x^{d_i})_{1 \leq i \leq \binom{n+d-1}{n-1}}$ les monômes de \mathcal{A}_d , triés par ordre décroissant selon l'ordre monomial w . Alors, pour $f_1, \dots, f_s \in \mathcal{A}$ des polynômes homogènes, $|f_i| = d_i$, et pour $d \in \mathbb{N}$, nous définissons la matrice de Macaulay en degré d (selon w), notée $Mac_d(f_1, \dots, f_s)$, comme la matrice à coefficients dans \mathcal{K} dont les lignes sont les $x^{\alpha_{1,1}} f_1, \dots, x^{\alpha_{1, \binom{n+d-d_1-1}{n-1}}} f_1, x^{\alpha_{2,1}} f_2, \dots, x^{\alpha_{s, \binom{n+d-d_s-1}{n-1}}} f_s$, écrit dans la base $B_{n,d}$. Les $x^{\alpha_{i,1}} < \dots < x^{\alpha_{i, \binom{n+d-d_i-1}{n-1}}}$ sont les monômes de degré $n + d - d_i - 1$. La j -ème colonne de cette matrice correspond aux j -ème monôme de $B_{n,d}$, x^{d_j} . Graphiquement, on peut la représenter ainsi :

$$\begin{array}{c}
 x^{\alpha_{1,1}} f_1 \\
 \vdots \\
 x^{\alpha_{1, \binom{n+d-d_1-1}{n-1}}} f_1 \\
 x^{\alpha_{2,1}} f_2 \\
 \vdots \\
 x^{\alpha_{s, \binom{n+d-d_s-1}{n-1}}} f_s
 \end{array}
 \begin{array}{c}
 x^{d_1} > \dots > \dots > x^{d_{\binom{n+d-1}{n-1}}} \\
 \left[\begin{array}{c} \\ \\ * \\ \\ \end{array} \right]
 \end{array}$$

Nous pouvons remarquer que, en prenant l'image à gauche de $Mac_d(f_1, \dots, f_s)$, on a

$$Im(Mac_d(f_1, \dots, f_s)) = I \cap \mathcal{A}_d.$$

4. Cette partie doit beaucoup aux lectures des thèses de Huot [Huo13] et Svartz [Sva14].

6. Algorithmes classiques pour le calcul de bases de Gröbner

De plus le premier terme non nul de chaque ligne correspond au coefficient dominant (pour w) du polynôme correspondant à cette ligne.

Cette seconde remarque nous permet de caractériser les bases de Gröbner en termes purement matriciels

Théorème 6.2.2 (Lazard [Laz83]). *Les polynômes homogènes f_1, \dots, f_s forment une base de Gröbner de l'idéal I qu'il engendrent si et seulement si, pour tout $d \in \mathbb{N}$, $Mac_d(f_1, \dots, f_s)$ contient une base échelonnée de $Im(Mac_d(f_1, \dots, f_s))$.*

Pour la notion de *base échelonnée*, nous utilisons la définition suivante :

Définition 6.2.3. Soit $M \in M_{n,m}(\mathcal{K})$ une matrice à coefficients dans \mathcal{K} ayant $n \in \mathbb{N}$ lignes et $m \in \mathbb{N}$ colonnes. Un ensemble L_{u_1}, \dots, L_{u_s} de lignes de M , pour des $u_i \in \llbracket 1, n \rrbracket$, est une base échelonnée de $Im(M)$ si $Vect(L_{u_1}, \dots, L_{u_s}) = Im(M)$ et que la sous-matrice de M restreinte aux lignes u_i est sous forme échelonnée (en lignes) à permutation près.

Exemple 6.2.4. Les lignes $(1, 3)$ forment une base échelonnée de $Im(M)$ pour la matrice $M \in Mat_{3,4}(\mathbb{Q})$ suivante :

$$M = \begin{bmatrix} 0 & 0 & 4 & 0 \\ 0 & 4 & 5 & 2 \\ 0 & 2 & 0 & 1 \end{bmatrix}.$$

Étant donnée une matrice, l'algorithme d'échelonnement en lignes gaussien 1.2.3 calcule une forme échelonnée de cette matrice. Il est alors aisé d'imaginer qu'étant donné $f = (f_1, \dots, f_s)$ des polynômes homogènes, calculer des formes échelonnées de toutes les matrices de Macaulay $Mac_d(f)$ pour d jusqu'à un degré D assez grand serait suffisant pour obtenir des bases échelonnées de toutes ces matrices et ce faisant, une base de Gröbner de l'idéal $\langle f \rangle$. Comme il n'est cependant pas aisé d'estimer jusqu'à quel degré D il faut échelonner les matrices de Macaulay, on définit la notion de D -base de Gröbner.

Définition 6.2.5. Soit I un idéal de \mathcal{A} , w un ordre monomial sur \mathcal{A} et D un entier. Alors (g_1, \dots, g_l) est une D -base de Gröbner de I si pour tout $f \in I$, homogène de degré au plus D , il existe $1 \leq i \leq l$ tels que, par rapport à w , $LM(g_i)$ divise $LM(f)$.

Il est alors naturel d'en déduire l'algorithme de Lazard de calcul de D -bases de Gröbner.

Algorithme 6.2.6 : L'algorithme de Lazard

entrée : $F = (f_1, \dots, f_s) \in \mathcal{A}^s$, polynômes homogènes de degré $d_1 \leq \dots \leq d_s$, et $D \in \mathbb{N}$, un ordre monomial w .

sortie : $(g_1, \dots, g_k) \in \mathcal{A}^k$, une D -base de Gröbner de $\langle F \rangle$.

début

```

     $G \leftarrow \{\}$  ;
    pour  $d \in \llbracket 0, D \rrbracket$  faire
         $M \leftarrow Mac_d(f_1, \dots, f_s)$  ;
        Calculer  $\widetilde{M}$ , une forme échelonnée de  $M$  ;
        Ajouter à  $G$  les polynômes des lignes de  $\widetilde{M}$  dont le monôme de tête n'est divisible par
        aucun monôme de tête d'un élément de  $G$ .;
    Retourner  $G$  ;
```

Nous pouvons estimer la complexité du calcul d'une D -base de Gröbner par l'algorithme de Lazard :

Théorème 6.2.7. *Si $(f_1, \dots, f_s) \in \mathcal{K}[X_1, \dots, X_n]^s$, sont des polynômes homogènes de degré $d_1 \leq \dots \leq d_s$, et $D \in \mathbb{N}$, alors on peut calculer une D -base de Gröbner de $\langle f_1, \dots, f_s \rangle$, pour un ordre monomial w donné, grâce à l'algorithme de Lazard en $O\left(s \binom{n+D}{D}^3\right)^5$ opérations dans \mathcal{K} pour $D \rightarrow +\infty$.*

5. Nous pourrions conserver la majoration plus fine : $O\left(s D \binom{n+D-1}{D}^3\right)$.

Démonstration. Nous utilisons le fait (voir Proposition 1.2.4) que l'échelonnement d'une matrice ayant n_{lignes} lignes et n_{col} colonnes est en $O(n_{lignes} \times n_{col}^2)$. La matrice $Mac_d(f_1, \dots, f_s)$ a $\sum_{i=1}^s \binom{n+d+d_i-1}{n-1}$ lignes et $\binom{n+d-1}{n-1}$ colonnes. On peut majorer le nombre de lignes par $s \binom{n+d-1}{n-1}$. Ceci amène une complexité totale en $O\left(\sum_{d=0}^D s \binom{n+d-1}{n-1}^3\right)$. Par convexité on peut écrire $\sum_{d=0}^D \binom{n+d-1}{n-1}^3 \leq \left(\sum_{d=0}^D \binom{n+d-1}{n-1}\right)^3$. Ce dernier terme peut se majorer par $\left((D+1) \binom{n+D-1}{n-1}\right)^3$, et on peut finalement utiliser que $D+1 \leq n+D$ pour obtenir une complexité en $O\left(s \binom{n+D}{D}^3\right)$. \square

Pour ce qui est du calcul d'une base de Gröbner, il n'est pas aisé d'estimer à partir de quel D une D -base de Gröbner est une base de Gröbner. Néanmoins, on dispose du résultat suivant lorsqu'on considère une suite régulière :

Proposition 6.2.8. *Si $(f_1, \dots, f_n) \in \mathcal{A}^n$ est une suite régulière de polynômes homogènes, toutes les D -bases de Gröbner sont des bases de Gröbner dès que $D \geq \sum_i (|f_i| - 1) + 1$.*

Démonstration. Ceci vient du fait que pour $(f_1, \dots, f_n) \in \mathcal{A}^n$ une suite régulière de polynômes homogènes, on peut montrer en regardant la série de Hilbert de $I = \langle f_1, \dots, f_n \rangle$ que tout monôme de degré D est dans I . Nous renvoyons à [BFS14] pour plus de précisions. \square

Le critère F5

L'algorithme de Lazard fournit une manière plutôt aisée de comprendre et d'analyser le calcul d'une base de Gröbner. Cependant, la complexité de cet algorithme est souvent trop grande pour qu'il soit utilisable en pratique. En particulier, une partie importante du temps de calcul est passé à réduire à zéro des lignes des matrices de Macaulay, dont le rang est souvent petit comparé au nombre de lignes. Le critère F5 de Faugère fournit une amélioration décisive de l'algorithme de Lazard en donnant un moyen de supprimer *a priori* des lignes des Matrices de Macaulay qui se réduiraient à zéro. Ce critère supprime la plupart de celles-ci, et même toutes dans le cas où les polynômes en entrée forment une suite régulière. Pour une introduction à ce critère, nous renvoyons à l'article original de Faugère [Fau02] ou à l'étude d'ensemble des algorithmes fondés sur le critère F5 effectuée par Eder et Faugère [EF14]. Nous résumons l'énoncé de ce critère avec le théorème suivant :

Théorème 6.2.9 (Critère F5). *Soit $(f_1, \dots, f_s) \in \mathcal{A}^s$ des polynômes homogènes, et $d \in \mathbb{N}$. Si l'on note pour $i \in \llbracket 1, s \rrbracket$, $I_i = \langle f_1, \dots, f_i \rangle$, alors :*

$$Im(Mac_d(f_1, \dots, f_s)) = Vect(\{x^\alpha f_k, t.q. 1 \leq k \leq s, |x^\alpha f_k| = d \text{ et } x^\alpha \notin LM(I_{k-1})\}).$$

Démonstration. Notons $V = Vect(\{x^\alpha f_k, t.q. 1 \leq k \leq s, |x^\alpha f_k| = d \text{ et } x^\alpha \notin LM(I_{k-1})\})$. Soit $k \in \llbracket 1, s \rrbracket$ et x^α monôme de degré $d - |f_k|$. Supposons que $x^\alpha \in LM(I_{k-1})$. Alors, il existe $r \in \mathcal{A}$ et $a_1, \dots, a_{k-1} \in \mathcal{A}$ tels que $x^\alpha + r = \sum_{i=1}^{k-1} a_i f_i \in I_{k-1}$ et $LM(r) < x^\alpha$. Nous en déduisons que $x^\alpha f_k = (-r)f_k + \sum_{i=1}^{k-1} (a_i f_k) f_i$. Ainsi, $x^\alpha f_k$ est combinaison linéaire des $x^\beta f_k$ avec $x^\beta < x^\alpha$ et des $x^\gamma f_i$ avec $i < k$. Ceci suffit pour montrer le résultat par récurrence. \square

Nous pouvons alors étudier les sous-matrices de la matrice de Macaulay dont on aurait supprimé les lignes inutiles par le critère 6.2.9 :

Définition 6.2.10. Soit $(f_1, \dots, f_s) \in \mathcal{A}^s$ des polynômes homogènes, et $d \in \mathbb{N}$. Nous notons par $\overline{Mac_d}(f_1, \dots, f_s)$ la sous-matrice de $Mac_d(f_1, \dots, f_s)$ définie par les lignes $x^\alpha f_k$ telles que $x^\alpha \notin LM(I_{k-1})$.

Lorsque les polynômes en entrée (f_1, \dots, f_s) forment une suite régulière, ces matrices $\overline{Mac_d}(f_1, \dots, f_s)$ vérifient la propriété suivante, qui permettra leur étude ensuite lorsqu'on travaille à précision finie :

Théorème 6.2.11. *Si (f_1, \dots, f_s) est une suite régulière, alors $\overline{Mac_d}(f_1, \dots, f_s)$ est injective (à gauche). En d'autres termes, si on l'échelonne en lignes, aucune ligne ne sera réduite à zéro.*

Démonstration. Supposons que la ligne correspondant à $x^\alpha f_k$ se réduise à zéro lors d'un échelonnement en lignes de $\overline{Mac_d(f_1, \dots, f_s)}$. Dans ce cas, il existe $l \in \llbracket 1, s \rrbracket$ et a_1, \dots, a_l polynômes homogènes de degré $d - d_1, \dots, d - d_l$ tels que $a_l \neq 0$ et $x^\alpha f_k = \sum_{i=1}^l a_i f_i$. En conséquence, il existe un $t \in \llbracket 1, s \rrbracket$ et b_1, \dots, b_t polynômes homogènes de degré $d - d_1, \dots, d - d_t$ tels que $b_t \neq 0$ et $\sum_{i=1}^t b_i f_i = 0$. Comme (f_1, \dots, f_t) est une suite régulière, f_t n'est pas diviseur de zéro dans $\mathcal{A} / \langle f_1, \dots, f_{t-1} \rangle$. En conséquence, $b_t \in \langle f_1, \dots, f_{t-1} \rangle$ et $LM(b_t) \in I_{t-1}$. Or, du fait du critère F5, il n'y a pas de ligne $LM(b_t) f_t$ dans $\overline{Mac_d(f_1, \dots, f_s)}$, et aucune combinaison linéaire sur les lignes $x^\alpha f_i$ de $\overline{Mac_d(f_1, \dots, f_s)}$ ne peut donc écrire $\sum_{i=1}^t b_i f_i = 0$. Nous montrons donc par l'absurde qu'en échelonnant $\overline{Mac_d(f_1, \dots, f_s)}$ aucune ligne ne peut être réduite à zéro. Ceci équivaut à l'injectivité (à gauche) de cette matrice. \square

Cependant, les théorèmes précédents ne nous disent pas comment construire ces matrices $\overline{Mac_d(f_1, \dots, f_s)}$. Ce sera fait avec l'algorithme F5-Matriciel.

Un premier algorithme F5-Matriciel

Le but de l'algorithme F5-Matriciel est de permettre l'utilisation du critère F5 lors du calcul de bases de Gröbner. Une première idée est la suivante : calculer des formes échelonnées des matrices de Macaulay $\overline{Mac_d(f_1, \dots, f_i)}$ itérativement en d et i , en utilisant le calcul déjà effectué de $\overline{Mac_{d-d_i}(f_1, \dots, f_{i-1})}$ pour appliquer le critère F5 lors de la construction de $\overline{Mac_d(f_1, \dots, f_i)}$ pour ainsi obtenir $\overline{Mac_d(f_1, \dots, f_s)}$.

Le pseudo-code suivant donne une première version de cet algorithme :

Algorithme 6.2.12 : Un premier algorithme F5-Matriciel

entrée : $F = (f_1, \dots, f_s) \in \mathcal{K}[X_1, \dots, X_n]^s$, polynômes homogènes de degré $d_1 \leq \dots \leq d_s$, et $D \in \mathbb{N}$,

un ordre monomial w .

sortie : $(g_1, \dots, g_k) \in \mathcal{A}^k$, une D -base de Gröbner de $\langle F \rangle$.

début

```

   $G \leftarrow \{ \} ;$ 
  pour  $d \in \llbracket 0, D \rrbracket$  faire
     $\widetilde{\mathcal{M}}_{d,0} := \emptyset ;$ 
    pour  $i \in \llbracket 1, s \rrbracket$  faire
       $\mathcal{M}_{d,i} := \widetilde{\mathcal{M}}_{d,i-1} ;$ 
      pour  $\alpha$  tel que  $|\alpha| + d_i = d$  faire
        si  $x^\alpha$  n'est pas le terme de tête d'une ligne de  $\widetilde{\mathcal{M}}_{d-d_i,i-1}$  alors
          Ajouter  $x^\alpha f_i$  à  $\mathcal{M}_{d,i} ;$ 
        Calculer  $\widetilde{\mathcal{M}}_{d,i}$ , la forme échelonnée de  $\mathcal{M}_{d,i} ;$ 
        Ajouter à  $G$  les polynômes des lignes de  $\widetilde{\mathcal{M}}_{d,i}$  avec un nouveau monôme de tête ;
        /* Ceci correspond aux lignes dont le monôme de tête n'est pas dans  $\langle LM(G) \rangle$  */
    Retourner  $G ;$ 
```

Nous avons alors le résultat suivant concernant la correction, la terminaison et la complexité de cet algorithme.

Proposition 6.2.13. *L'algorithme F5-Matriciel prenant en entrée $(f_1, \dots, f_s) \in \mathcal{A}$, $d \in \mathbb{N}$ et w un ordre monomial termine et calcule une D -base de Gröbner de l'idéal $\langle f_1, \dots, f_s \rangle$. La complexité est en $O\left(s^2 \binom{n+D}{D}^3\right)$ ou $O\left(s \binom{n+D}{D}^3\right)$ lorsque (f_1, \dots, f_s) est une suite régulière, lorsque $D \rightarrow +\infty$.⁶*

Démonstration. Pour la terminaison, comme on ne considère que des boucles **pour**, il n'y a rien à démontrer. Pour ce qui est de la correction, il suffit de montrer par récurrence sur d et i que $\text{Im}(\mathcal{M}_{d,i}) = I_i \cap \mathcal{A}_d$. Mais ceci est clair grâce au théorème 6.2.9 et au fait que l'échelonnement

6. Il est possible de remplacer $\binom{n+D}{D}^3$ par $D \binom{n+D-1}{D}^3$ dans les majorations précédentes.

d'une matrice de Macaulay fournit bien les termes de tête de l'idéal correspondant à ses lignes, au degré correspondant à la matrice de Macaulay.

Pour ce qui est de la complexité, la preuve est analogue à celle du Théorème 6.2.7 dans le cas général. Pour le cas régulier, il suffit de remarquer que toutes les matrices considérées ont moins de lignes que de colonnes et sont de rang plein. \square

Remarque 6.2.14. L'estimation de la complexité dans la proposition précédente est plutôt naïve, et ne permet pas de comparer correctement les algorithmes de Lazard et F5-Matriciel. La version de l'algorithme F5-Matriciel donnée dans la Sous-Section suivante donnera une complexité nettement meilleure.

Exemple 6.2.15. Nous appliquons l'Algorithme 6.2.12 sur $F = (f_1, f_2, f_3) \in \mathbb{Q}[x, y, z]$ pour grevlex avec $f_1 = 2x + z$, $f_2 = x^2 + y^2 - 2z^2$ et $f_3 = 4y^2 + yz + 8z^2$. Nous prenons $D = 3$, la borne de Macaulay. Nous partons de $G = (f_1, f_2, f_3)$.

En degré 1, nous obtenons $\mathcal{M}_{1,1} = \mathcal{M}_{1,2} = \mathcal{M}_{1,3}$ qui sont $f_1 \begin{array}{c|ccc} & x & y & z \\ \hline & 2 & 0 & 1 \end{array}$ et qui est déjà sous forme échelonnée.⁷

En degré 2, $\mathcal{M}_{2,1} = \widetilde{\mathcal{M}}_{2,1}$ est

$$\begin{array}{c|ccccc} & x^2 & xy & y^2 & xz & yz & z^2 \\ \hline zf_1 & & & & 2 & & 1 \\ yf_1 & & 2 & & & 1 & \\ xf_1 & 2 & & & 1 & & \end{array}.$$

Ensuite, $\mathcal{M}_{2,2}$ est

$$\begin{array}{c|ccccc} & x^2 & xy & y^2 & xz & yz & z^2 \\ \hline zf_1 & & & 2 & & & 1 \\ yf_1 & & 2 & & & 1 & \\ xf_1 & 2 & & & 1 & & \\ f_2 & 1 & & 1 & & -2 & \end{array} \quad \text{et } \widetilde{\mathcal{M}}_{2,2} \text{ est}$$

$$\begin{array}{c|ccccc} & x^2 & xy & y^2 & xz & yz & z^2 \\ \hline zf_1 & & & & 2 & & 1 \\ yf_1 & & 2 & & & 1 & \\ xf_1 & 0 & & -2 & 1 & & 4 \\ f_2 & 1 & & 1 & & -2 & \end{array}.$$

Ainsi, nous ajoutons à G $f_4 = -2y^2 + xz + 4z^2$.

De même : $\mathcal{M}_{2,3}$ est

$$\begin{array}{c|ccccc} & x^2 & xy & y^2 & xz & yz & z^2 \\ \hline zf_1 & & & & 2 & & 1 \\ yf_1 & & 2 & & & 1 & \\ xf_1 & 0 & & -2 & 1 & & 4 \\ f_2 & 1 & & 1 & & -2 & \\ f_3 & & & 4 & & 1 & 8 \end{array} \quad \text{et } \widetilde{\mathcal{M}}_{2,3} \text{ est}$$

$$\begin{array}{c|ccccc} & x^2 & xy & y^2 & xz & yz & z^2 \\ \hline zf_1 & & & & 2 & & 1 \\ yf_1 & & 2 & & & 1 & \\ xf_1 & 0 & & -2 & 1 & & 4 \\ f_2 & 1 & & 1 & & -2 & \\ f_3 & & & 0 & & 1 & 15 \end{array}.$$

À G est ajouté $f_5 = yz + 15z^2$.

En degré 3, nous obtenons $\mathcal{M}_{3,1} = \widetilde{\mathcal{M}}_{3,1}$ qui est la matrice suivante :

$$\begin{array}{c|cccccccc} & x^3 & x^2y & xy^2 & y^3 & x^2z & xyz & y^2z & xz^2 & yz^2 & z^3 \\ \hline z^2f_1 & & & & & & & & 2 & & 1 \\ y^2f_1 & & & & & & 2 & & & 1 & \\ x^2f_1 & & & & & 2 & & & 1 & & \\ y^2f_1 & & & 2 & & & & 1 & & & \\ xyf_1 & & 2 & & & & 1 & & & & \\ x^2f_1 & 2 & & & & 1 & & & & & \end{array}.$$

7. Remarquons que la convention que nous avons prise pour la notion de forme échelonnée ne demande pas de normalisation, i.e. que sur chaque ligne, le premier coefficient non nul soit 1.

6. Algorithmes classiques pour le calcul de bases de Gröbner

Grâce au critère F5, nous pouvons écarter xf_2 et seulement ajouter zf_2 et yf_2 pour définir

$$\mathcal{M}_{3,2} \text{ qui correspond à } \begin{array}{c} z^2 f_1 \\ y^z f_1 \\ x^z f_1 \\ y^2 f_1 \\ xy f_1 \\ x^2 f_1 \\ z f_2 \\ y f_2 \end{array} \left| \begin{array}{cccccccc} x^3 & x^2 y & xy^2 & y^3 & x^2 z & xyz & y^2 z & xz^2 & yz^2 & z^3 \\ & & & & & & & 2 & & 1 \\ & & & & & 2 & & & 1 & \\ & & & & 2 & & & 1 & & \\ & & 2 & & & & 1 & & & \\ & 2 & & & & 1 & & & & \\ 2 & & & & 1 & & & & & \\ & & & & 1 & & 1 & & & -2 \\ & 1 & & 1 & & & & & -2 & \end{array} \right|. \text{ Nous obtenons ainsi}$$

$$\widetilde{\mathcal{M}}_{3,2} \text{ comme ceci : } \begin{array}{c} z^2 f_1 \\ y^z f_1 \\ x^z f_1 \\ y^2 f_1 \\ xy f_1 \\ x^2 f_1 \\ z f_2 \\ y f_2 \end{array} \left| \begin{array}{cccccccc} x^3 & x^2 y & xy^2 & y^3 & x^2 z & xyz & y^2 z & xz^2 & yz^2 & z^3 \\ & & & & & & & 2 & & 1 \\ & & & & & 2 & & & 1 & \\ & & & & 0 & & -2 & 1 & & \\ & & 2 & & & & 1 & & & \\ & 0 & & -2 & & 1 & & & 4 & \\ 2 & & & & 1 & & & & & \\ & & & & 1 & & 1 & & & -2 \\ & 1 & & 1 & & & & & -2 & \end{array} \right|. \text{ À } G, \text{ nous ajoutons}$$

$f_6 = -2y^3 + xyz + 4yz^2$ et $f_7 = -2y^2z + xz^2$.

Enfin, à nouveau grâce au critère F5, nous pouvons écarter xf_3 et définir $\mathcal{M}_{3,3}$ comme

$$\begin{array}{c} z^2 f_1 \\ y^z f_1 \\ x^z f_1 \\ y^2 f_1 \\ xy f_1 \\ x^2 f_1 \\ z f_2 \\ y f_2 \\ z f_3 \\ y f_3 \end{array} \left| \begin{array}{cccccccc} x^3 & x^2 y & xy^2 & y^3 & x^2 z & xyz & y^2 z & xz^2 & yz^2 & z^3 \\ & & & & & & & 2 & & 1 \\ & & & & & 2 & & & 1 & \\ & & & & 0 & & -2 & 1 & & \\ & & 2 & & & & 1 & & & \\ & 0 & & -2 & & 1 & & & 4 & \\ 2 & & & & 1 & & & & & \\ & & & & 1 & & 1 & & & -2 \\ & 1 & & 1 & & & & & -2 & \\ & & & & & & 4 & & 1 & 8 \\ & & & 4 & & & 1 & & 8 & \end{array} \right|.$$

$$\begin{array}{c}
z^2 f_1 \\
y^z f_1 \\
x^z f_1 \\
y^2 f_1 \\
xy f_1 \\
x^2 f_1 \\
z f_2 \\
y f_2 \\
z f_3 \\
y f_3
\end{array}
\begin{array}{c}
x^3 \quad x^2 y \quad xy^2 \quad y^3 \quad x^2 z \quad xyz \quad y^2 z \quad xz^2 \quad yz^2 \quad z^3 \\
\left| \begin{array}{cccccccccc}
& & & & & & & 2 & & 1 \\
& & & & & 2 & & & 1 & \\
& & & 0 & & -2 & 1 & & & \\
& & 2 & & & & 1 & & & \\
0 & & -2 & & 1 & & & & 4 & \\
2 & & & & 1 & & & & & \\
& & & & 1 & & 1 & & & -2 \\
1 & & 1 & & & & & & -2 & \\
& & & 0 & & 0 & & 1 & 7 & \\
& & & 0 & & & 0 & 0 & 105 &
\end{array} \right|
\end{array}$$

Finale-ment, nous obtenons $\widetilde{\mathcal{M}}_{3,3}$ comme :
Ainsi, les derniers polynômes ajoutés à G sont $f_8 = yz^2 + 7z^3$ et $f_9 = 105z^3$. Nous obtenons finale-ment comme base de Gröbner minimale (f_1, f_3, f_5, f_9) , qui, à renormalisation des coefficients de tête près, est réduite.

Un algorithme F5-matriciel avec signatures

Il n'est pas évident, vu la manière dont nous avons présenté l'algorithme F5-Matriciel précédente, de savoir s'il gagne en complexité par rapport à l'algorithme de Lazard. Néanmoins, nous allons voir qu'il est possible d'accélérer ce qui concerne la réduction des matrices en utilisant le fait que lorsqu'on construit $\mathcal{M}_{d,i}$ à partir de $\widetilde{\mathcal{M}}_{d,i-1}$, cette dernière matrice est déjà sous forme échelonnée. Ceci nous donnera une version plus rapide de l'algorithme F5-Matriciel.

Étiquettes et signatures À cet effet, nous introduisons les concepts d'étiquette et de signature de polynômes. Cette dernière servira lors des échelonnements matriciels et suffira pour appliquer le critère F5.

Définition 6.2.16. Soit $(f_1, \dots, f_s) \in \mathcal{A}^s$, un *polynôme étiqueté* est un couple (u, P) avec $u = (l_1, \dots, l_s) \in \mathcal{A}^s$, $P \in \mathcal{A}$ et $\sum_{i=1}^s l_i f_i = P$. Nous appelons u l'*étiquette* de ce polynôme étiqueté.

Notons (e_1, \dots, e_s) la base canonique de \mathcal{A}^s . Si $u = (l_1, \dots, l_i, 0, \dots, 0)$ avec $l_i \neq 0$, alors la *signature* du polynôme étiqueté (u, p) , notée $\text{sign}((u, p))$, est $(LM(l_i), i)$.

Définition 6.2.17. L'ensemble des signatures $\{\text{monomes de } \mathcal{A}\} \times \{1, \dots, s\}$ peut être muni d'un ordre total défini de la manière suivante : $(x^\alpha, i) \leq (x^\beta, k)$ si $i < k$, ou $x^\alpha \leq x^\beta$ et $i = k$.

Les signatures sont compatibles avec les opérations usuelles sur les polynômes :

Proposition 6.2.18. Soit (u, p) un polynôme étiqueté, $(x^\alpha, i) = \text{sign}((u, l))$ et soit x^β un monôme de \mathcal{A} . Alors

$$\text{sign}((x^\beta u, x^\beta p)) = (x^\alpha x^\beta, i).$$

Si (v, q) est un autre polynôme étiqueté tel que $\text{sign}((v, q)) < \text{sign}((u, p))$, et si $\mu \in \mathcal{K}$, alors $\text{sign}((u + \mu v, p + \mu q)) = \text{sign}((u, p))$.

Un algorithme d'échelonnement sans choix du pivot À partir de maintenant et durant toute cette Sous-Section, nous attachons aux matrices de Macaulay les étiquettes et les signatures de chaque ligne. Nous demanderons de plus que les lignes soient triées par signature croissante (la première ligne ayant la plus petite signature). Lorsque nous effectuerons des opérations sur les lignes d'une matrice de Macaulay étiquetée, les opérations seront répercutées sur les étiquettes et signatures correspondants. Ce seront ces signatures qui permettront d'appliquer le critère F5. Pour

6. Algorithmes classiques pour le calcul de bases de Gröbner

cela, nous introduisons un algorithme d'échelonnement en lignes sans choix du pivot qui préserve les signatures.

Algorithme 6.2.19 : L'algorithme d'échelonnement sans choix du pivot

entrée : M , une matrice de Macaulay avec étiquettes et signatures, de degré d sur \mathcal{A} , ayant n_{lignes} lignes et n_{col} colonnes. Nous demandons que les lignes soient ordonnées par ordre croissant de signature.

sortie : \widetilde{M} , une forme échelonnée à permutation près de M

début

$\widetilde{M} \leftarrow M$;

si M n'a pas de coefficient non-nul **alors**

Retourner \widetilde{M} ;

sinon

Trouver le plus petit i tel que $M_{i,1} \neq 0$;

Par pivot avec la ligne i , éliminer les coefficients sur la première colonne des lignes sous la ligne i , et appeler le résultat \widetilde{M} ;

Procéder récursivement sur la sous-matrice de $\widetilde{M}_{j \geq 2}$ privée de la ligne i ;

Retourner \widetilde{M} ;

Nous remarquons qu'à la sortie de l'algorithme, il existe une matrice unipotente triangulaire inférieure L , une matrice de permutation P , tels que $\widetilde{M} = LMP$, \widetilde{M} est sous forme échelonnée (en lignes). De plus, comme à chaque ligne L_i , nous avons au plus ajouté une combinaison linéaire de lignes L_j , avec $j < i$, qui ont une signature strictement plus petite que L_i , les signatures sont conservées. En outre,

Proposition 6.2.20. *Pour tout $1 \leq i \leq n_{row}(M)$, si j est l'indice de la i -ème ligne de \widetilde{M} , alors $\widetilde{M}_{i,j}x^{mon_j}$ est le terme de tête du polynôme correspondant à cette ligne.*

Ces remarques justifient le nom d'algorithme d'échelonnement sans choix du pivot, de même que le fait que cet algorithme calcule les termes de têtes de $Vect(Lignes(M))$. Enfin, comme les signatures restent inchangées durant le calcul de l'échelonnement sans choix du pivot, il est suffisant de seulement noter les signatures de chaque ligne sur les matrices de Macaulay étiquetées. Ce sera le choix fait dans toute la suite de cette Sous-Section.

Un algorithme F5-Matriciel plus rapide Nous montrons maintenant qu'il est possible d'appliquer l'échelonnement sans choix du pivot dans l'algorithme F5-matriciel. Ceci permettra d'utiliser le fait que lorsqu'on construit les matrices de Macaulay, les lignes déjà échelonnées ne nécessitent plus aucun travail.

Tout d'abord, le critère F5 avec signature peut être utilisé :

Proposition 6.2.21. *Soit (u, f) un polynôme homogène étiqueté de degré d . Supposons que $sign(u) = (x^\alpha, i)$, avec $1 < i \leq s$ et $x^\alpha \in I_{i-1}$. Alors,*

$$x^\alpha f_i \in Vect \left(\{x^\beta f_k, |x^\beta f_k| = d, \text{ et } (x^\beta, k) < (x^\alpha, i)\} \right).$$

En conséquence, si (u, f) est un polynôme homogène de degré d avec $sign(u) = x^\alpha e_i$ et $x^\alpha \notin LM(I_{i-1})$. Alors f peut s'écrire $f = x^\alpha f_i + g$, avec

$$g \in Vect \left(\{x^\beta f_k, |x^\beta f_k| = d, \text{ et } (x^\beta, k) < (x^\alpha, i)\} \right).$$

Nous pouvons alors appliquer cette version du critère F5 et l'échelonnement sans choix du pivot

pour proposer l'algorithme F5-Matriciel avec signatures suivant :

Algorithme 6.2.22 : Algorithme F5-Matriciel avec signatures

entrée : $F = (f_1, \dots, f_s) \in \mathcal{A}^s$, homogènes de degrés respectifs d_1, \dots, d_s , et $D \in \mathbb{N}$.

sortie : $(g_1, \dots, g_k) \in \mathcal{A}^k$, une D -base de Gröbner de $\langle F \rangle$.

début

```

   $G \leftarrow F$  ;
  pour  $d \in \llbracket 0, D \rrbracket$  faire
     $\widetilde{\mathcal{M}}_{d,0} := \emptyset$  ;
    pour  $i \in \llbracket 1, s \rrbracket$  faire
       $\mathcal{M}_{d,i} := \widetilde{\mathcal{M}}_{d,i-1}$  ;
      pour  $L$  une ligne de  $\widetilde{\mathcal{M}}_{d-1,i}$  faire
        pour  $x \in \{X_1, \dots, X_n\}$  faire
           $(x^\alpha, k) := \text{sign}(xL)$  ;
          si  $k = i$  et  $x^\alpha$  n'est pas le terme d'une ligne de  $\widetilde{\mathcal{M}}_{d-d_i,i-1}$  et  $\mathcal{M}_{d,i}$  n'a pas
            déjà une ligne de signature  $(x^\alpha, i)$  alors
            Ajouter  $xL$  à  $\mathcal{M}_{d,i}$  ;
        Trier les lignes de  $\mathcal{M}_{d,i}$  par signature croissante ;
        Calculer  $\widetilde{\mathcal{M}}_{d,i}$ , la forme échelonnée sans choix du pivot de  $\mathcal{M}_{d,i}$  ;
        Ajouter à  $G$  les lignes avec un nouveau monôme de tête ;
    Retourner  $G$  ;

```

Correction La première chose à montrer est que lors de la construction des matrices de Macaulay lors de l'exécution de l'algorithme, les deux propriétés suivantes sont satisfaites : $\text{Im}(\mathcal{M}_{d,i}) = I_i \cap \mathcal{A}_d$ et pour tout monôme x^α de degré $d - d_i$ tel que $x^\alpha \notin \text{LM}(I_{i-1})$, $\mathcal{M}_{d,i}$ a une ligne ayant pour signature $x^\alpha e_i$. Ceci peut se prouver par récurrence sur d et i .

Maintenant, comme l'algorithme d'échelonnement sans choix du pivot calcule une base échelonnée de $\mathcal{M}_{d,i}$, de même que lors de l'algorithme F5-Matriciel précédent, l'algorithme F5-Matriciel avec signature calcule une D -base de Gröbner.

Concernant les suites régulières, nous avons à nouveau, et pour les mêmes raisons qu'au Théorème 6.2.11, la propriété suivante :

Proposition 6.2.23. *Si $(f_1, \dots, f_s) \in \mathcal{A}^s$ est une suite régulière de polynômes homogènes. Alors les $\mathcal{M}_{d,i}$ sont injectives.*

Complexité La différence principale du point de vue de la complexité entre les Algorithmes 6.2.12 et 6.2.22 est que dans le second cas, le calcul de la forme échelonnée sans choix du pivot $\mathcal{M}_{d,i+1}$ prend en compte le fait que le calcul a déjà été fait pour $\mathcal{M}_{d,i}$, i.e. les premières lignes de $\mathcal{M}_{d,i+1}$ sont déjà sous-forme échelonnée avec les bons termes de tête, et aucun nouveau calcul n'est à ajouter. En conséquence, la complexité du calcul d'une D -base de Gröbner (f_1, \dots, f_s) est la suivante :

Proposition 6.2.24. *L'algorithme F5-Matriciel prenant en entrée $(f_1, \dots, f_s) \in \mathcal{A}$, $d \in \mathbb{N}$ et w un ordre monomial termine et calcule une D -base de Gröbner de l'idéal $\langle f_1, \dots, f_s \rangle$ avec une complexité en $O\left(s \binom{n+D}{D}^3\right)$ lorsque $D \rightarrow +\infty$. Si (f_1, \dots, f_s) est une suite régulière, la complexité est en $O\left(\binom{n+D}{D}^3\right)$.⁸*

Démonstration. Du fait de l'usage de l'algorithme d'échelonnement sans choix du pivot, le coût des échelonnements des matrices $\mathcal{M}_{d,1}, \dots, \mathcal{M}_{d,s}$ est exactement celui de l'échelonnement de $\overline{\text{Mac}_d(f_1, \dots, f_s)}$, soit $O\left(\sum_{i=1}^s \binom{n+d+d_i-1}{n-1} \binom{n+d-1}{n-1}^2\right)$ ou $O\left(\binom{n+d-1}{n-1}^3\right)$ dans le cas régulier. En

8. Il est là encore possible de remplacer $\binom{n+D}{D}^3$ par $D \binom{n+D-1}{D}^3$ dans les majorations précédentes.

conséquence, lorsqu'on somme sur d de 0 à D , on obtient une complexité en $O\left(s\binom{n+D}{D}^3\right)$, ou $O\left(\binom{n+D}{D}^3\right)$ dans le cas régulier, par les même majorations que dans la preuve du Théorème 6.2.7. \square

Un exemple

Exemple 6.2.25. Nous reprenons l'exemple 6.2.15, en appliquant cette fois un algorithme F5-Matriciel avec signatures. Nous débutons à nouveau avec $G = (f_1, f_2, f_3)$.

Nous obtenons les résultats suivants. En degré 1, il n'y a pas de différence.

$$\begin{array}{c} x^2 \quad xy \quad y^2 \quad xz \quad yz \quad z^2 \\ \begin{array}{l} zf_1 \\ yf_1 \\ xf_1 \\ f_2 \end{array} \left| \begin{array}{cccccc} & & 2 & & & 1 \\ & 2 & & & 1 & \\ 2 & & & 1 & & \\ 1 & & 1 & & & -2 \end{array} \right| \end{array} \text{ et}$$

En degré 2, $\mathcal{M}_{2,1} = \widetilde{\mathcal{M}}_{2,1}$ reste identique. Ensuite, $\mathcal{M}_{2,2}$ est

$$\begin{array}{c} x^2 \quad xy \quad y^2 \quad xz \quad yz \quad z^2 \\ \begin{array}{l} zf_1 \\ yf_1 \\ xf_1 \\ f_2 \end{array} \left| \begin{array}{cccccc} & & 2 & & & 1 \\ & 2 & & & 1 & \\ 2 & & & 1 & & \\ 0 & 0 & 1 & -1/2 & & -2 \end{array} \right| \end{array} \text{ alors } \widetilde{\mathcal{M}}_{2,2} \text{ est}$$

. Nous ajoutons $f_4 = y^2 - \frac{1}{2}xz - 2z^2$ à G .

$$\begin{array}{c} x^2 \quad xy \quad y^2 \quad xz \quad yz \quad z^2 \\ \begin{array}{l} zf_1 \\ yf_1 \\ xf_1 \\ f_2 \\ f_3 \end{array} \left| \begin{array}{cccccc} & & & 2 & & 1 \\ & 2 & & & 1 & \\ 0 & & -2 & 1 & & 4 \\ 1 & & 1 & & & -2 \\ & & 4 & & 1 & 8 \end{array} \right| \end{array} \text{ et } \widetilde{\mathcal{M}}_{2,3} \text{ est}$$

De même : $\mathcal{M}_{2,3}$ est

À G est ajouté $f_5 = yz + 15z^2$.

En degré 3, nous obtenons de même $\mathcal{M}_{3,1} = \widetilde{\mathcal{M}}_{3,1}$ qui est la matrice suivante :

$$\begin{array}{c} x^3 \quad x^2y \quad xy^2 \quad y^3 \quad x^2z \quad xyz \quad y^2z \quad xz^2 \quad yz^2 \quad z^3 \\ \begin{array}{l} z^2f_1 \\ y^2f_1 \\ x^2f_1 \\ y^2f_1 \\ xyf_1 \\ x^2f_1 \end{array} \left| \begin{array}{ccccccccc} & & & & & & 2 & & & 1 \\ & & & & 2 & & & & 1 & \\ & & & 2 & & & & 1 & & \\ & & 2 & & & & 1 & & & \\ & 2 & & & & 1 & & & & \\ 2 & & & & 1 & & & & & \end{array} \right| \end{array}$$

Grâce au critère F5, nous pouvons écarter la signature xf_2 et seulement ajouter les multiples de lignes de $\widetilde{\mathcal{M}}_{2,2}$ ayant pour signatures zf_2 et yf_2 pour définir $\mathcal{M}_{3,2}$ qui correspond à

$$\begin{array}{c}
x^3 \quad x^2y \quad xy^2 \quad y^3 \quad x^2z \quad xyz \quad y^2z \quad xz^2 \quad yz^2 \quad z^3 \\
\begin{array}{l}
z^2f_1 \\
y^zf_1 \\
x^zf_1 \\
y^2f_1 \\
xyf_1 \\
x^2f_1 \\
zf_2 \\
yf_2
\end{array}
\left| \begin{array}{cccccccccc}
& & & & & & & 2 & & 1 \\
& & & & & 2 & & & 1 & \\
& & & 2 & & & & 1 & & \\
& & 2 & & & & 1 & & & \\
& 2 & & & & 1 & & & & \\
2 & & & & 1 & & & & & \\
& & & & & & 1 & & -1/2 & -2 \\
& & & 1 & & -1/2 & & & -2 &
\end{array} \right|
\end{array}$$

. Remarquons en particulier $\widetilde{\mathcal{M}}_{3,2}$ s'obtient alors directement sans calcul.

Enfin, à nouveau grâce au critère F5, nous pouvons écarter la signature xf_3 et définir $\mathcal{M}_{3,3}$ comme

$$\begin{array}{c}
x^3 \quad x^2y \quad xy^2 \quad y^3 \quad x^2z \quad xyz \quad y^2z \quad xz^2 \quad yz^2 \quad z^3 \\
\begin{array}{l}
z^2f_1 \\
y^zf_1 \\
x^zf_1 \\
y^2f_1 \\
xyf_1 \\
x^2f_1 \\
zf_2 \\
yf_2 \\
zf_3 \\
yf_3
\end{array}
\left| \begin{array}{cccccccccc}
& & & & & & & 2 & & 1 \\
& & & & & 2 & & & 1 & \\
& & & 2 & & & & 1 & & \\
& & 2 & & & & 1 & & & \\
& 2 & & & & 1 & & & & \\
2 & & & & 1 & & & & & \\
& & & & & & 1 & & -1/2 & -2 \\
& & & 1 & & -1/2 & & & -2 & \\
& & & & & & & & 1 & 15 \\
& & & & & & 1 & & 15 &
\end{array} \right|
\end{array}$$

. Finalement, nous obtenons $\widetilde{\mathcal{M}}_{3,3}$ comme :

$$\begin{array}{c}
x^3 \quad x^2y \quad xy^2 \quad y^3 \quad x^2z \quad xyz \quad y^2z \quad xz^2 \quad yz^2 \quad z^3 \\
\begin{array}{l}
z^2f_1 \\
y^zf_1 \\
x^zf_1 \\
y^2f_1 \\
xyf_1 \\
x^2f_1 \\
zf_2 \\
yf_2 \\
zf_3 \\
yf_3
\end{array}
\left| \begin{array}{cccccccccc}
& & & & & & & 2 & & 1 \\
& & & & & 2 & & & 1 & \\
& & & 2 & & & & 1 & & \\
& & 2 & & & & 1 & & & \\
& 2 & & & & 1 & & & & \\
2 & & & & 1 & & & & & \\
& & & & & & 1 & & -1/2 & -2 \\
& & & 1 & & -1/2 & & & -2 & \\
& & & & & & & & 1 & 15 \\
& & & & & & 0 & & 0 & 230.5
\end{array} \right|
\end{array}$$

. À G est ajouté $f_6 = 230.5z^3$.

En conclusion (f_1, f_3, f_5, f_6) est la base de Gröbner minimale correspondante, et elle est réduite à renormalisation des coefficients de tête près.

6.2.2. L'algorithme FGLM

Le principe

L'algorithme de changement de base FGLM a été introduit par Faugère, Gianni, Lazard et Mora en 1993 dans [FGLM93]. Pour un idéal $I \subset \mathcal{A}$ de dimension zéro, il permet par des calculs d'algèbre linéaire dans l'espace vectoriel \mathcal{A}/I de calculer une base de Gröbner G_2 pour un ordre monomial

\leq_2 connaissant une base G_1 pour un premier ordre monomial \leq .

L'idée principale est de considérer les monômes x^α de \mathcal{A} par ordre croissant selon \leq_2 , et de calculer leurs images $NF_{\leq}(x^\alpha)$ par la projection sur le quotient \mathcal{A}/I grâce à G_1 . Chaque relation linéaire entre les $NF_{\leq}(x^\alpha)$ donne un polynôme de I . En calculant dans le bon ordre ces relations, on obtient une base de Gröbner G_2 .

Dans toute cette Sous-Section, I est un idéal de dimension zéro, \leq et \leq_2 sont des ordres monomiaux et G une base de Gröbner de I pour \leq .

Calculs dans le quotient L'algorithme FGLM utilise de manière essentielle l'arithmétique dans le quotient \mathcal{A}/I . Nous présentons ici une manière efficace de travailler dans celui-ci.

Pour cela, nous définissons d'abord une présentation adaptée du quotient :

Définition 6.2.26. Nous notons $B_{\leq} = \{x^\alpha | x^\alpha \notin LM_{\leq}(I)\}$ la base canonique (par rapport à \leq) du \mathcal{H} -espace vectoriel \mathcal{A}/I . Ses éléments sont ordonnés par ordre croissant pour \leq .

Nous pouvons préciser quel est le premier terme de B_{\leq} :

Remarque 6.2.27. Comme I est un idéal de dimension zéro, I est un idéal strict et $1 \notin I$. En conséquence $1 \notin LM_{\leq}(I)$ et 1 est le premier monôme de B_{\leq} .

Nous définissons aussi la forme normale d'un polynôme, ou projeté sur \mathcal{A}/I .

Définition 6.2.28. Soit $P \in \mathcal{A}$. Nous définissons $NF_{\leq}(P)$, forme normale de P pour I , comme le reste de $P \bmod I$ écrit dans la base B_{\leq} .

Afin de calculer aisément dans \mathcal{A}/I et de calculer aisément les formes normales, nous introduisons les matrices de multiplication par les variables de \mathcal{A} . Elles permettront de calculer la projection $NF_{\leq}(P)$ d'un $P \in \mathcal{A}$ par de simples multiplications matrices-vecteurs et additions.

Définition 6.2.29 (Matrices de multiplication). Pour $i \in \llbracket 1, n \rrbracket$, nous notons T_i la matrice, écrite dans la base B_{\leq} , de l'endomorphisme de \mathcal{A}/I défini par la multiplication par x_i . Nous l'appelons la i -ème matrice de multiplication de \mathcal{A}/I .

Si l'on connaît les T_i , nous pouvons en effet calculer dans \mathcal{A}/I en utilisant l'algorithme suivant :

Algorithme 6.2.30 : Calcul d'une forme normale par les matrices de multiplication

entrée : Un idéal $I \subset \mathcal{A}$ de dimension zéro, ses matrices de multiplication T_1, \dots, T_n par rapport à un ordre monomial \leq et un polynôme $f = \sum_{\alpha} a_{\alpha} x^{\alpha} \in \mathcal{A}$.

sortie : $NF_{\leq}(f)$ la forme normale de f pour I .

début

 Soit $\mathbf{1} = (1, \dots, 0)$ l'écriture dans $(\mathcal{A}/I, B_{\leq})$ du monôme 1 ;
 Retourner $\sum_{\alpha} a_{\alpha} T_1^{\alpha_1} \dots T_n^{\alpha_n} \mathbf{1}$;

Exemple 6.2.31. Soit $I = \langle f_1, f_2, f_3 \rangle$ avec $f_1 = x + z$, $f_2 = x^2 + y^2 + z^2$ et $f_3 = y^2 + yz$ dans $\mathbb{Q}[x, y, z]$. Pour grevlex, la base de Gröbner réduite de I est $(x + z, y^2 + 2z^2, yz - 2z^2, z^3)$, et ainsi, $B_{\text{grevlex}} = (1, z, y, z^2)$. Les matrices de multiplications sont alors :

$$M_1 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & 0 \end{pmatrix}$$

$$M_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 2 & 2 & 0 \end{pmatrix}$$

$$M_3 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 0 \end{pmatrix}$$

Calcul effectif des matrices de multiplication Pour calculer efficacement les matrices de multiplications, nous avons besoin de mieux comprendre le comportement des monômes et de leurs formes normales lorsqu'on effectue des multiplications. À cet effet, nous introduisons les concepts d'escalier et de bord :

Définition 6.2.32. Nous définissons l'escalier et le bord de I pour \leq par :

- l'escalier $\mathcal{E}_{\leq}(I)$ est B_{\leq} ;
- le bord $\mathcal{B}_{\leq}(I)$ est l'ensemble $\{x_i\epsilon \mid i \in \llbracket 1, n \rrbracket \text{ et } \epsilon \in \mathcal{E}_{\leq}(I)\} \setminus \mathcal{E}_{\leq}(I)$.

Dans [FGLM93], une caractérisation est alors donnée pour le bord de I :

Proposition 6.2.33. Soit $x^\alpha \in \mathcal{B}_{\leq}(I)$. Alors x^α vérifie une et une seule des propositions suivantes :

- Pour tout $i \in \llbracket 1, n \rrbracket$ tel que x_i divise x^α , $\frac{x^\alpha}{x_i} \in \mathcal{E}_{\leq}(I)$. Ceci arrive si et seulement si x^α est terme de tête d'un élément de la base de Gröbner réduite de I pour \leq .
- x^α peut s'écrire $x_i x^\beta$ pour un certain $i \in \llbracket 1, n \rrbracket$ et $x^\beta \in \mathcal{B}_{\leq}(I)$.

Ceci va être suffisant pour calculer les matrices de multiplication. En effet, pour les calculer, il suffit de calculer tous les $NF_{\leq}(x_i\epsilon)$ pour $i \in \llbracket 1, n \rrbracket$ et $\epsilon \in \mathcal{E}_{\leq}(I)$. Soit L la liste de ces $x_i\epsilon$, triée par ordre croissant pour \leq et sans répétition. Alors, pour un élément $u \in L$, il y a seulement trois possibilités :

1. $u \in \mathcal{E}_{\leq}(I)$ et dans ce cas, il n'y a pas de calcul à faire puisque $NF_{\leq}(u) = u$;
2. u est terme de tête d'un élément g de la base de Gröbner réduite de I pour \leq et dans ce cas, on obtient directement $NF_{\leq}(u) = u - g$;
3. d'après la Proposition 6.2.33, la seule autre possibilité est qu'il existe $v \in \mathcal{B}_{\leq}(I)$ et $i \in \llbracket 1, n \rrbracket$ tels que $u = x_i v$. Dans ce cas, si l'on parcourt L itérativement, $NF_{\leq}(v)$ a déjà été calculé, et il existe des $a_k \in \mathcal{K}$ et $\epsilon_k \in \mathcal{E}_{\leq}(I)$ tels que $NF_{\leq}(v) = \sum_k a_k \epsilon_k$. De plus, nous avons $LM(NF_{\leq}(v)) < v$. Nous en déduisons que pour tous les k tels que $a_k \neq 0$ dans l'écriture précédente, $x_i \epsilon_k < u$. Ainsi, les $NF_{\leq}(x_i \epsilon_k)$ ont eux aussi déjà été calculés. Nous en déduisons que dans la base B_{\leq} , $NF_{\leq}(u) = \sum_k a_k T_i[\cdot, \epsilon_k]$ avec $T_i[\cdot, k]$ la k -ème colonne de T_i (qui par définition, est l'écriture de $T_i \epsilon_k$ dans B_{\leq}).

Nous en déduisons un algorithme de calcul des matrices de multiplication :

Algorithme 6.2.34 : Calcul des matrices de multiplication

entrée : Une base de Gröbner réduite G d'un idéal $I \subset \mathcal{A}$ de dimension zéro et de degré δ pour un ordre monomial \leq , $B_{\leq} = (1 = \epsilon_1 \leq \epsilon_2 \leq \dots \leq \epsilon_{\delta})$ la base canonique de \mathcal{A}/I pour \leq .

sortie : Les matrices de multiplication T_i pour I et \leq .

début

```

pour  $i \in \llbracket 1, n \rrbracket$  faire
   $T_i := 0_{\delta \times \delta}$  ;
 $L := [x_i \epsilon_k \mid i \in \llbracket 1, n \rrbracket \text{ et } \epsilon_k \in B_{\leq}]$ , triée par ordre croissant et sans répétition ;
pour  $u \in L$  faire
  si  $u \in \mathcal{E}_{\leq}(I)$  alors
     $T_i[u, u/x_i] = 1$  pour tout  $i$  tel que  $x_i \mid u$  ;
    /* La colonne indexée par  $u$  est nulle, sauf pour son coefficient indexé par  $u/x_i$  */
  sinon si  $u = LM(g)$  pour un certain  $g \in G$  alors
     $g$  s'écrit  $u + \sum_{k=1}^{\delta} a_k \epsilon_k$  ;
     $T_i[\cdot, u/x_i] := -{}^t(a_1, \dots, a_{\delta})$  pour tout  $i$  tel que  $x_i \mid u$  ;
  sinon
    Trouver le plus petit  $x_j$  pour  $\leq$  tel que  $x_j \mid u$  ;
    Soit  $v = u/x_j$  ;
    Trouver  $\epsilon$  et  $l$  tels que  $v = x_l \epsilon$  ;
     $V := T_l[\cdot, v]$  (cette colonne contient  $NF_{\leq}(v)$ ) ;
     $W := T_j V$  ( $W$  est le vecteur correspondant à la forme normale
     $NF_{\leq}(x_j v) = NF_{\leq}(u)$ ) ;
     $T_i[\cdot, u/x_i] := W$  pour tout  $i$  tel que  $x_i \mid u$  ;
  Retourner  $T_1, \dots, T_n$  ;
```

Nous avons alors la proposition suivante :

Proposition 6.2.35. *Étant donnés une base de Gröbner réduite G d'un idéal de dimension zéro et de degré δ $I \subset \mathcal{A}$ pour un ordre monomial \leq , et $B_{\leq} = (1 = \epsilon_1 \leq \epsilon_2 \leq \dots \leq \epsilon_{\delta})$ la base canonique de \mathcal{A}/I pour \leq , l'Algorithme 6.2.34 calcule les matrices de multiplication dans \mathcal{A}/I . La complexité est en $O(n\delta^3)$ opérations arithmétiques dans \mathcal{K} .*

Démonstration. Avec ce qui précède, correction et terminaison sont claires. Pour ce qui est de la complexité, L contient au plus $n\delta$ monômes, et pour chaque passage dans la boucle, au plus une multiplication matrice-vecteur. Cette dernière coûte $O(\delta^2)$ opérations dans \mathcal{K} . Nous pouvons en déduire le résultat. \square

Une première version de l'algorithme Maintenant que nous avons présenté tous les outils nécessaires pour présenter l'algorithme FGLM, nous pouvons en donner l'idée principale : pour calculer une base de Gröbner de I pour \leq_2 , il s'agit de calculer $\mathcal{E}_{\leq_2}(I)$ et $\mathcal{B}_{\leq_2}(I)$, et de déduire cette base de Gröbner du calcul de $\mathcal{B}_{\leq_2}(I)$. Pour cela, on utilise le fait que les éléments de $\mathcal{E}_{\leq_2}(I)$ forment une base de \mathcal{A}/I tandis que pour les $x^{\alpha} \in \mathcal{B}_{\leq_2}(I)$, on a l'alternative issue de la Proposition 6.2.33 suivante : soit x^{α} est monôme de tête dans la base de Gröbner réduite de I pour \leq_2 et dans ce cas, $NF_{\leq}(x^{\alpha}) \in Vect(NF_{\leq}(\mathcal{E}_{\leq_2}(I)))$, soit x^{α} est le multiple d'un monôme de tête d'un élément de la base de Gröbner réduite de I pour \leq_2 .

Si l'on regarde les monômes x^{α} par ordre croissant en commençant par 1 (qui n'est pas dans I puisque I est de dimension zéro), alors pour le cas où $x^{\alpha} \in \mathcal{B}_{\leq_2}(I)$, les monômes de $\mathcal{E}_{\leq_2}(I)$ plus petits que x^{α} suffisent à écrire $NF_{\leq}(x^{\alpha})$ dans $Vect(NF_{\leq}(\mathcal{E}_{\leq_2}(I)))$, et dans le second cas, si x^{α} est le multiple d'un monôme de tête x^{β} de la base de Gröbner réduite de I pour \leq_2 alors $x^{\beta} \leq_2 x^{\alpha}$ et x^{β} aura précédemment été traité selon le premier cas.

Nous pouvons en déduire l'algorithme suivant, première version, simplifiée, de l'algorithme FGLM :

Algorithme 6.2.36 : Algorithme FGLM simplifié

entrée : Une base de Gröbner réduite G_1 pour un ordre monomial \leq d'un idéal $I \subset \mathcal{A}$ de dimension zéro et de degré δ , $B_{\leq} = (1 = \epsilon_1 \leq \epsilon_2 \leq \dots \leq \epsilon_\delta)$ la base canonique de \mathcal{A}/I pour \leq .
Un ordre monomial \leq_2 .
sortie : Une base de Gröbner G_2 de I pour \leq_2 .

début

```

Calculer les matrices de multiplications  $T_1, \dots, T_n$  pour  $I$  et  $\leq$  avec l'Algorithme 6.2.34 ;
 $B_2 := \{1\}$  ; //  $B_2$  représente  $\mathcal{E}_{\leq_2}$ , en cours de construction
 $\mathbf{v}[1] := {}^t(1, \dots, 0)$  ; //  $\mathbf{v}$  représente les  $NF_{\leq}(x^\alpha)$  pour  $x^\alpha \in B_2$ 
 $LT := \emptyset$  ; //  $LT$  est  $\mathcal{B}_{\leq_2}(I)$ , en construction
 $G_2 := \emptyset$  ;
 $L := \{x_i \epsilon_j \mid 1 \leq i \leq n, \epsilon_j \in B_2 \text{ tels que } x_i \epsilon_j \notin B_2 \cup LT\}$  ;
/*  $L$  est le bord de  $B_2$ , moins  $LT$  et ses multiples ; il décrit les monômes pouvant donner un
   monôme de tête pour  $\leq_2$ , et est inclus dans  $\mathcal{B}_{\leq_2} \cup \mathcal{E}_{\leq_2}$  */
tant que  $L \neq \emptyset$  faire
     $x^\alpha := \min L$  par rapport à  $\leq_2$  ;
    si il existe  $x^\beta \in LT$  tel que  $x^\beta$  divise  $x^\alpha$  alors
        |  $LT := LT \cup \{x^\alpha\}$ 
    sinon
        Trouver  $x_i$  et  $\epsilon$  tels que  $x^\alpha = x_i \epsilon$  ;
         $\mathbf{v}[x^\alpha] := T_i \cdot \mathbf{v}[\epsilon]$  ;
        si  $\mathbf{v}[x^\alpha]$  n'est pas dans  $\text{Vect}(\{\mathbf{v}[\epsilon] \mid \epsilon \in B_2\})$  alors
            |  $B_2 := B_2 \cup \{x^\alpha\}$  ;
        sinon
            |  $LT := LT \cup \{x^\alpha\}$  ;
            | Soit  $c_\epsilon \in \mathcal{K}$  tels que  $\mathbf{v}[x^\alpha] = \sum_{\epsilon \in B_2} c_\epsilon \mathbf{v}[\epsilon]$  ;
            |  $G_2 := G_2 \cup \{x^\alpha - \sum_{\epsilon \in B_2} c_\epsilon \epsilon\}$  ;
        |  $L := \{x_i \epsilon \mid 1 \leq i \leq n \text{ et } \epsilon \in B_2 \text{ tels que } x_i \epsilon \notin B_2 \cup LT\}$  ;

```

Nous avons alors le résultat suivant :

Proposition 6.2.37. *L'Algorithme 6.2.36 termine et calcule bien une base de Gröbner G_2 pour \leq_2 de l'idéal I donné en entrée.*

Démonstration. Pour montrer la correction et la terminaison, nous montrons d'abord qu'à chaque passage de la boucle **Tant que**, on a l'invariant suivant : $LT \subset \mathcal{B}_{\leq_2}(I)$, $B_2 \subset \mathcal{E}_{\leq_2}(I)$ et pour tout $\epsilon \in B_2$, $\mathbf{v}[\epsilon] = NF_{\leq}(\epsilon)$. La troisième partie est claire par l'Algorithme 6.2.30, tandis que les deux premières sont conséquences directes de la Proposition 6.2.33.

La terminaison est conséquence directe de cet invariant de boucle. En effet à chaque passage dans la boucle, la valeur de $\text{card}(\mathcal{B}_{\leq_2}(I) \setminus LT) + \text{card}(\mathcal{E}_{\leq_2}(I) \setminus B_2)$ décroît strictement.

Ainsi, l'algorithme termine et en sortie, $L = \mathcal{B}_{\leq_2}(I)$, $B_2 = \mathcal{E}_{\leq_2}(I)$. La correction est alors une conséquence de la Proposition 6.2.33. \square

Bien que cette version simplifiée de l'algorithme FGLM permette de comprendre le principe de l'algorithme, elle ne répond pas aux contraintes effectives sur la manière de tester efficacement si par exemple, la forme normale $\mathbf{v}[x^\alpha]$ du monôme x^α est dans $\text{Vect}(\{\mathbf{v}[\epsilon] \mid \epsilon \in B_2\})$. En particulier, nous ne pouvons pas analyser la complexité de cet algorithme sur cette version simplifiée.

Une version effective

Afin de proposer une version effective, implémentable et dont on pourrait analyser le temps de calcul, nous proposons une version plus précise de l'Algorithme 6.2.36.

Le principe est de regarder la matrice $M(\leq, \leq_2)$ ayant δ lignes et dont les colonnes (en quantité dénombrable) correspondent aux $NF_{\leq}(x^\alpha)$, écrits dans la base B_{\leq} , triés par ordre croissant pour x^α

selon \leq_2 . Nous remarquons que déterminer $\mathcal{E}_{\leq_2}(I)$ et $\mathcal{B}_{\leq_2}(I)$ revient à calculer une forme échelonnée réduite de $M(\leq, \leq_2)$. Pour cela, nous effectuons l'élimination avec pivot colonne par colonne : les colonnes où l'on trouve un pivot sont celles correspondant aux $x^\alpha \in \mathcal{E}_{\leq_2}(I)$. Lorsqu'on rencontre une colonne sans pivot, cela correspond à un $x^\alpha \in LM_{\leq_2}(I)$, et la forme échelonnée réduite suffit à trouver un g dont ce serait le monôme de tête. Enfin, en ignorant les colonnes correspondant à des x^β multiples des x^α du cas précédent, nous avons traité tous les cas. En particulier, considérer un nombre fini de colonnes suffit.

Voici une manière explicite de réaliser cet algorithme :

Algorithme 6.2.38 : Algorithme FGLM effectif

entrée : Une base de Gröbner réduite G_1 pour un ordre monomial \leq d'un idéal $I \subset \mathcal{A}$ de dimension zéro et de degré δ , $B_{\leq} = (1 = \epsilon_1 \leq \epsilon_2 \leq \dots \leq \epsilon_\delta)$ la base canonique de \mathcal{A}/I pour \leq .

Un ordre monomial \leq_2 .

sortie : Une base de Gröbner G_2 de I pour \leq_2 .

début

```

Calculer les matrices de multiplication  $T_1, \dots, T_n$  pour  $I$  et  $\leq$  avec l'Algorithme 6.2.34 ;
 $B_2 := \{1\}$  ; //  $B_2$  représente  $\mathcal{E}_{\leq_2}$ , en cours de construction
 $\mathbf{v} = [{}^t(1, \dots, 0)]$  ; //  $\mathbf{v}$  représente les  $NF_{\leq}(x^\alpha)$  pour  $x^\alpha \in B_2$ 
 $G_2 := \emptyset$  ;
 $L := \{(1, n), (1, n-1), \dots, (1, 1)\}$  ;
/* Les éléments de  $L$  sont des  $(l, k)$ , représentant le monôme  $x_k B_2[l]$ . Ils correspondent au bord
   de  $B_2$ , moins les multiples des éléments de  $LM_{\leq_2}(G_2)$  ;  $L$  décrit les monômes pouvant
   donner un monôme de tête pour  $\leq_2$ , et est inclus dans  $\mathcal{B}_{\leq_2} \cup \mathcal{E}_{\leq_2}$  */
 $Q := I_\delta$  ; //  $Q$  est la matrice, en construction, de changement de base de  $\mathcal{E}_{\leq}$  vers  $\mathcal{E}_{\leq_2}$ 
tant que  $L \neq \emptyset$  faire
     $m := L[1]$  ; supprimer  $m$  de  $L$  ;
     $j := m[1]$  ;  $i := m[2]$  ;
     $v := T_i \mathbf{v}[j]$  ;
     $s := \text{card}(B_2)$  ;
     $\lambda = {}^t(\lambda_1, \dots, \lambda_\delta) := Qv$  ;
    si  $\lambda_{s+1} = \dots = \lambda_\delta = 0$  alors
         $G_2 := G_2 \cup \{B_2[j]x_i - \sum_{l=1}^s \lambda_l B_2[l]\}$ 
    sinon
         $B_2 := B_2 \cup \{B_2[j]x_i\}$  ;
         $\mathbf{v} = \mathbf{v} \cup [v]$  ;
         $L := \text{TriCroissant}(L \cup [(s+1, l) | 1 \leq l \leq n], \leq_2)$  ;
        Enlever les répétitions dans  $L$  ;
         $Q := \text{Update}(Q, s, \lambda)$  ;
    Enlever de  $L$  tous les multiples de  $LM_{\leq_2}(G_2)$  ;
Retourner  $G_2$  ;
```

Lorsque nous écrivons $\text{TriCroissant}(L \cup [(s+1, l) | 1 \leq l \leq n], \leq_2)$, nous voulons parler du tri (par ordre croissant) selon \leq_2 des monômes $B_2[i]x_k$ correspondant aux couples (i, k) du tableau en argument de TriCroissant .

Dans cet algorithme, nous utilisons une sous-procédure, Update , qui correspond au pas d'échelonnement de la nouvelle colonne, dans le cas où elle est indépendante des précédentes. Voici à quoi

cet algorithme correspond :

Algorithme 6.2.39 : Update, échelonnement partiel

entrée : Un entier s , une matrice $Q \in \mathcal{K}^{\delta \times \delta}$ et un vecteur $\lambda \in \mathcal{K}^\delta$ tel que $\lambda[s+1 \leq i \leq \delta] \neq [0, \dots, 0]$.
sortie : Une matrice PQ où P est tel $Pe_i = e_i$ pour $i \in \llbracket 1, s \rrbracket$ et $PQ\lambda = e_{s+1}$.

début

```

   $l := \lambda$  ;
  Trouver le plus petit  $i$  dans  $\llbracket s+1, \delta \rrbracket$  tel que  $l[i] \neq 0$  ;
  /* Variante : celui donnant le maximum des  $|l[i]|$  */
   $Q := \text{MatPerm}(s+1, i) \cdot Q$  ;
   $l := \text{MatPerm}(s+1, i)l$  ;
   $Q := \text{MatDilat}(s+1, 1/l[s+1]) \cdot Q$  ;
   $l := \text{MatDilat}(s+1, 1/l[s+1])l$  ;
  pour  $j$  de 1 à  $\delta$  en évitant  $s+1$  faire
     $Q := \text{MatTransv}(s+1, j, l[j]) \cdot Q$  ;
     $l := \text{MatTransv}(s+1, j, l[j])l$  ;
  Retourner  $Q$  ;
  /* Remarquons que nous écrivons des produits matriciels pour effectuer nos opérations sur les
    lignes. On pourrait effectuer ces opérations autrement. */

```

Lemme 6.2.40. *L'Algorithme 6.2.39, prenant en entrée un entier s , une matrice $Q \in \mathcal{K}^{\delta \times \delta}$ et un vecteur $\lambda \in \mathcal{K}^\delta$ tel que $\lambda[s+1 \leq i \leq \delta] \neq [0, \dots, 0]$, calcule une matrice $P \in M_\delta(\mathcal{K})$ telle que $Pe_i = e_i$ pour $i \in \llbracket 1, s \rrbracket$ et $PQ\lambda = e_{s+1}$ et renvoie PQ . Sa complexité est en $O(\delta^2)$.*

Démonstration. Les matrices T des opérations élémentaires que l'on effectue sur Q et l satisfont bien toutes $Te_i = e_i$ pour $i \in \llbracket 1, s \rrbracket$, et sont choisies pour transformer l en e_{s+1} . Le résultat est alors clair. Comme on effectue $\delta + 1$ opérations élémentaires sur les lignes de Q qui est une matrice $\delta \times \delta$, et aussi sur les lignes de l , la complexité est bien en $O(\delta^2)$. \square

Nous pouvons maintenant énoncer le théorème suivant concernant une réalisation effective de FGLM :

Théorème 6.2.41. *Soit G_1 une base de Gröbner réduite pour un ordre monomial \leq d'un idéal $I \subset \mathcal{A}$ de dimension zéro et de degré δ . Soit $B_{\leq} = (1 = \epsilon_1 \leq \epsilon_2 \leq \dots \leq \epsilon_\delta)$ la base canonique de \mathcal{A}/I pour \leq . Soit \leq_2 un ordre monomial. Alors l'Algorithme 6.2.38 appliqué à ces entrées termine et renvoie une base de Gröbner G_2 de I pour \leq_2 . La complexité est en $O(n\delta^3)$ opérations arithmétiques sur \mathcal{K} .*

Démonstration. L'Algorithme 6.2.38 est essentiellement une variante de l'Algorithme 6.2.36. Correction et terminaison peuvent se faire de manière similaire.

Nous montrons simplement que la condition $\lambda_{s+1} = \dots = \lambda_\delta = 0$ correspond exactement à $NF_{\leq}(B_2[j]x_i)$ dans $\text{Vect}(NF_{\leq}(B_2))$. Pour cela, nous utilisons l'invariant de la boucle **tant que** suivant : en chaque entrée dans la boucle, Q est inversible et vérifie $Q\mathbf{v}[j] = e_j$ pour tout $j \in \llbracket 1, s \rrbracket$, où $s := \text{card}(B_2)$. Cet invariant est une conséquence directe de la preuve de correction de l'Algorithme 6.2.39. Remarquons aussi qu'avec la correction de l'Algorithme 6.2.30, on a aussi comme invariant de boucle $\mathbf{v}[j] = NF_{\leq}(B_2[j])$ pour tout $j \in \llbracket 1, s \rrbracket$. Mais alors, si $NF_{\leq}(m) = NF_{\leq}(B_2[j]x_i) \in \text{Vect}(B_2)$, comme $Q(\text{Vect}(NF_{\leq}(B_2))) = \text{Vect}(\{e_1, \dots, e_s\})$, on en déduit que $\lambda_{s+1} = \dots = \lambda_\delta = 0$. Réciproquement, si $\lambda_{s+1} = \dots = \lambda_\delta = 0$, $Qv \in \text{Vect}(Q(NF_{\leq}(B_2)))$ et comme Q est inversible, le résultat s'en déduit.

À partir de cet invariant, il est aisé de montrer correction et terminaison comme pour l'Algorithme 6.2.36.

Pour ce qui est du temps de calcul, nous remarquons tout d'abord que l'on passe au plus $O(n\delta)$ fois dans la boucle **tant que** puisque $\delta + n\delta$ majore le cardinal de $\mathcal{E}_{\leq_2}(I) \cup \mathcal{B}_{\leq_2}(I)$. Les opérations arithmétiques effectuées dans une boucle **tant que** sont au plus des produits matrices-vecteurs ou effectués dans l'appel à *Update*, qui est en $O(\delta^2)$ opérations arithmétiques. En conclusion, la complexité de l'usage de FGLM est en $O(n\delta^3)$ opérations arithmétiques sur \mathcal{K} . \square

6. Algorithmes classiques pour le calcul de bases de Gröbner

Nous illustrons maintenant cet algorithme à travers une continuation de l'exemple 6.2.31.

Exemple 6.2.42. Soit $I = \langle f_1, f_2, f_3 \rangle$ avec $f_1 = x + z$, $f_2 = x^2 + y^2 + z^2$ et $f_3 = y^2 + yz$ dans $\mathbb{Q}[x, y, z]$. Pour grevlex, la base de Gröbner réduite de I est $(x + z, y^2 + 2z^2, yz - 2z^2, z^3)$, et ainsi,

$B_{\text{grevlex}} = (1, z, y, z^2)$. Les matrices de multiplications sont alors $M_1 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & 0 \end{pmatrix}$, $M_2 =$

$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 2 & 2 & 0 \end{pmatrix}$ et $M_3 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 0 \end{pmatrix}$. Nous allons construire une base de Gröbner pour I pour

l'ordre $\text{lex}(z > y > x)$. En appliquant l'algorithme FGLM, nous trouvons $B_{\text{lex}(z > y > x)} = (1, x, x^2, y)$,

$\mathbf{v} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$, et $Q = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$.

Illustrons une des étapes de l'algorithme : après l'ajout de y à $B_{\text{lex}(z > y > x)}$, \mathbf{v} et Q ne seront plus modifiés et L correspond à $(xy, x^2y, y^2, z, xz, x^2z, yz)$. xy est obtenu comme $y \times x$, ou plus

précisément : $v = M_2 \cdot \begin{pmatrix} 0 \\ -1 \\ 0 \\ 0 \end{pmatrix}$. Nous avons alors $v = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -2 \end{pmatrix}$ et $Qv = \begin{pmatrix} 0 \\ 0 \\ -2 \\ 0 \end{pmatrix}$. Comme $Q\mathbf{v} = Id$,

nous avons bien Qv dans $\text{Im}(Q\mathbf{v})$, et nous obtenons $xy + 2x^2$ dans I .

En sortie de l'algorithme, nous obtenons $G_2 = (x^3, xy + 2x^2, x^2y, z + x, x^2 + xz, x^2z, y^2 + 2x^2, yz - 2y^2)$. Ceci permet bien d'obtenir la base de Gröbner réduite de I pour l'ordre $\text{lex}(z > y > x)$: $(x^3, xy + 2x^2, y^2 + 2x^2, z + x)$.

Et si l'on veut résoudre plus rapidement un système polynomial ?

Dans cette Sous-sous-section, nous présentons quelques variantes de l'algorithme FGLM qui ont été développés avec pour but de diminuer la complexité du calcul d'une base de Gröbner pour un ordre lexicographique dans certains particuliers (génériques en un certain sens). Pour ceux-ci, la résolution du système est ramenée complètement à un problème univarié, et nous allons voir que la complexité peut être abaissée de $O(n\delta^3)$ jusqu'à $O(\delta^3) + O(n\delta^2)$ opérations arithmétiques dans le corps de base. Ces algorithmes sont d'abord issus des travaux de Faugère et Mou dans [FM11, FM13, Mou13]. Nous n'en présentons qu'une version simplifiée, où par exemple, contrairement à ces articles, nous n'utilisons pas le caractère éventuellement creux des matrices qui apparaissent dans l'algorithme. D'autres variations sur l'algorithme FGLM ont été écrites dans la continuité des travaux précédents, avec notamment l'application d'algorithmes rapides en algèbre linéaire et l'étude du cas particulier du changement d'ordre de grevlex vers lex, voir : [FGHR13, FGHR14, Huo13].

Position générale et représentation univariée Afin d'étudier plus précisément la résolution de systèmes polynomiaux, la notion d'idéal en position générale⁹ et de représentation univariée a été introduite :

Définition 6.2.43. Un idéal I de \mathcal{A} , de dimension zéro et de degré δ est dit mettre les variables en **position générale** s'il existe $h_1, \dots, h_n \in \mathcal{K}[T]$ des polynômes, avec $\deg h_n = \delta$, tels que l'on ait l'isomorphisme de \mathcal{K} -algèbres suivant :

$$\begin{array}{ccc} \mathcal{A}/I = \mathcal{K}[X_1, \dots, X_n]/I & \rightarrow & \mathcal{K}[T]/\langle h_n \rangle \\ X_1, \dots, X_{n-1} & \mapsto & h_1(T), \dots, h_{n-1}(T) \\ X_n & \mapsto & T. \end{array}$$

Nous écrirons parfois simplement que I est en position générale.

⁹ L'auteur remercie Marc Giusti de lui avoir donné la référence [GH91] pour cette traduction à la notion d'idéal en *shape position*.

Définition 6.2.44. Un idéal I de \mathcal{A} , de dimension zéro et de degré δ , admet une **représentation univariée** (pour l'ordre lexicographique avec $X_1 \geq \dots \geq X_n$) si sa base de Gröbner réduite pour l'ordre lex est de la forme $(X_1 - h_1(X_n), \dots, X_{n+1} - h_{n+1}(X_{n-1}), h_n(X_n))$, avec $\deg(h_n) = \delta$ et $\deg(h_i) < \delta$ pour $i \in \llbracket 1, n-1 \rrbracket$. Nous disons que les h_i forment une représentation univariée de I .

Il est clair que vues les définitions précédentes, un idéal de dimension zéro est en position générale si et seulement si il admet une représentation univariée.

L'intérêt de l'étude de telles représentations est le suivant : étant donnée une représentation univariée d'un idéal en position générale, le calcul des points de la variété algébrique correspondante est entièrement déterminé par le calcul des solutions du polynôme univarié h_n .

Les idéaux en position générale, souvent appelés en *shape position* dans la littérature, ainsi que leur caractérisation ont été abondamment étudiés. Nous renvoyons à [GM89], [LL91], [GH91] et [BMMT94] pour plus de détails. Citons en particulier le résultat classique suivant sur la manière de se ramener à un idéal en position générale par changement de variables linéaire :

Proposition 6.2.45. Soit I un idéal de \mathcal{A} radiciel, de dimension zéro. Supposons que $\text{car}(\mathcal{K}) = 0$. Alors, il existe un ouvert de Zariski $U \subset GL_n(\mathcal{K})$ tel que pour tout $M \in U$, le changement de variable linéaire défini par U définit un idéal $M \cdot I$ en position générale.

Il reste maintenant à expliquer comment calculer efficacement une représentation univariée d'un idéal en position générale.

Calcul des h_i La forme particulière de la base de Gröbner pour l'ordre lexicographique d'un idéal en position générale permet de simplifier les calculs pour calculer cette base. En effet, nous pouvons énoncer la proposition suivante :

Proposition 6.2.46. Soit M la matrice dont les colonnes sont les $NF_{\leq}(1), \dots, NF_{\leq}(x_n^{\delta-1})$ écrits dans la base B_{\leq} . Notons $h_i(X) = -\sum_{j=0}^{\delta-1} a_{i,j} X^j$ pour $1 \leq i \leq n-1$ et $h_n(X) = X^n - \sum_{j=0}^{\delta-1} a_{n,j} X^j$. Alors, si l'on note $A_i = {}^t(a_{i,1}, \dots, a_{i,\delta-1})$ pour $1 \leq i \leq n$, on a $NF_{\leq}(x_i) = M A_i$ pour $1 \leq i \leq n-1$ et $NF_{\leq}(x_n^{\delta}) = M A_n$.

Cette proposition est essentiellement une réécriture de ce qui était utilisé dans la section précédente. Elle suggère une manière efficace de calculer les h_i : il suffit de calculer les $NF_{\leq}(x_n^j)$ et $NF_{\leq}(x_i)$ et de résoudre n systèmes linéaires avec la même matrice.

Nous pouvons remarquer qu'une autre caractérisation de h_n était possible :

Remarque 6.2.47. Soit I un idéal de \mathcal{A} , de dimension zéro et de degré δ en position générale. Soit $(x_1 - h_1(x_n), \dots, x_{n+1} - h_{n+1}(x_{n-1}), h_n(x_n))$. Soit \leq un ordre monomial. Soit T_1, \dots, T_n les matrices de multiplication définies par \leq . Alors $h_n = \chi_{T_n}$.

Démonstration. Soit I un idéal de \mathcal{A} , de dimension zéro et de degré δ en position générale. On a $\mathcal{E}_{lex}(I) = (1, x_n, \dots, x_n^{\delta-1})$. Du fait que la famille $\mathcal{E}_{lex}(I)$ est libre dans \mathcal{A}/I , on déduit que T_n n'a pas de polynôme annulateur de degré strictement inférieur à δ . En conséquence, nous avons bien $h_n = \chi_{T_n}$ (qui est même le polynôme minimal de T_n). \square

Modification de l'algorithme FGLM En conséquence de la Proposition 6.2.46, nous pouvons proposer la version suivante de l'algorithme FGLM pour calculer une représentation univariée d'un

6. Algorithmes classiques pour le calcul de bases de Gröbner

idéal en position générale.

Algorithme 6.2.48 : Algorithme FGLM pour un idéal en position générale

entrée : Une base de Gröbner réduite G_1 pour un ordre monomial \leq d'un idéal $I \subset \mathcal{A}$ de dimension zéro et de degré δ , $B_{\leq} = (1 = \epsilon_1 \leq \epsilon_2 \leq \dots \leq \epsilon_\delta)$ la base canonique de \mathcal{A}/I pour \leq .
 I est en position générale.
sortie : Une base de Gröbner G_2 de I pour \leq_{lex} .

début

Calculer les matrices de multiplications T_1, \dots, T_n pour I et \leq avec l'Algorithme 6.2.34 ;
 $G_2 := \emptyset$;
pour i de 1 à $n-1$ **faire**
 Calculer $\mathbf{y}[i] := T_i 1$;
 $\mathbf{z}[0] := 1$;
pour i de 1 à δ **faire**
 Calculer $\mathbf{z}[i] = T_n \mathbf{z}[i-1]$;
 $M := \text{Mat}_{B_{\leq}}(\mathbf{z}[0], \dots, \mathbf{z}[\delta-1])$;
pour i de 1 à $n-1$ **faire**
 Trouver U tel que $\mathbf{y}[i] = -M \cdot U$;
 $h_i(T) := \sum_{j=1}^{\delta-1} U[j] T^j$;
 Trouver U tel que $\mathbf{z}[\delta] = -M \cdot U$;
 $h_n(T) := T^\delta + \sum_{j=1}^{\delta-1} U[j] T^j$;
 Retourner $x_1 - h_1(x_n), \dots, x_{n-1} - h_{n-1}(x_n), h_n(x_n)$;

Nous avons alors la proposition suivante concernant correction et terminaison de l'Algorithme 6.2.48 :

Proposition 6.2.49. *Ayant en entrée une base de Gröbner réduite G_1 d'un idéal $I \subset \mathcal{A}$ de dimension zéro et de degré δ pour un ordre monomial \leq , et en supposant que I est en position générale, alors l'Algorithme 6.2.48 termine et renvoie bien une base de Gröbner G_2 de I pour \leq_{lex} . Le temps de calcul est en $O(\delta^3) + O(n\delta^2)$ plus $O(n\delta^3)$ pour le calcul des matrices de multiplication.*

Démonstration. Avec la définition d'idéal en position générale, la Proposition 6.2.46 et la correction de l'Algorithme FGLM (voir Algorithme 6.2.36), correction et terminaison sont claires. Nous remarquerons simplement que la matrice construite M est inversible du fait que $\mathcal{E}_{\leq_{lex}} = (1, x_n, \dots, x_n^{\delta-1})$. Pour ce qui est du temps de calcul, tout d'abord, le calcul des matrices de multiplication est en $O(n\delta^3)$. Les produits matrices-vecteurs sont en $O(\delta^2)$. Pour la résolution des systèmes linéaires, il suffit de calculer une fois l'inverse de la matrice M , ce qui est en $O(\delta^3)$, puis de faire des produits matrices-vecteurs. Ainsi, on effectue $O(n + \delta)$ produits matrices-vecteurs qui sont en $O(\delta^2)$. La complexité totale s'en déduit. \square

Nous pouvons néanmoins préciser la complexité en remarquant que les calculs des $T_i 1$ ($1 \leq i \leq n-1$) pourraient en fait être faits sans la connaissance de T_i ni opération arithmétique lorsqu'on considère un ordre monomial qui raffine le degré :

Proposition 6.2.50. *Si G est une base de Gröbner réduite d'un idéal I pour un ordre monomial \leq qui raffine le degré. Alors pour $1 \leq i \leq n-1$, $T_i 1$ peut être obtenu sans opération arithmétique.*

Démonstration. Il suffit de dissocier les deux cas, selon que $x_i \in LT(I)$ ou non. Dans le premier cas, il existe $g \in G$ tel que $LT(g) | x_i$. Comme on ne considère que des idéaux propres, on a $LT(g) = x_i$ et on peut lire $NF_{\leq}(x_i)$ sur g : si $g = x_i + \sum_{\alpha} a_{\alpha} x^{\alpha}$, alors $NF_{\leq}(x_i) = -\sum_{\alpha} a_{\alpha} x^{\alpha}$. Comme \leq raffine le degré, les x^{α} sont de degré 1. Dans le second cas, $NF_{\leq}(x_i) = x_i$. Ainsi, en prenant les x_i par ordre croissant selon \leq , nous pouvons bien en déduire les éléments de \mathcal{E}_{\leq} de degré 1 et les $T_i 1$. D'où le résultat. \square

Cas particulier de grevlex pour ordre monomial initial Dans le cas particulier où le premier ordre monomial \leq est un ordre grevlex, ce qui est souvent le cas en pratique, et si l'on est en présence

d'un idéal générique, au sens que nous allons définir, alors le calcul des matrices de multiplication est bien plus aisé. Ceci permettra de ramener la complexité de l'algorithme FGLM pour aller de grevlex vers lex à $O(\delta^3 + n\delta^2)$.

Pour cela nous pouvons nous appuyer sur la proposition suivante, énoncée par exemple dans [Huo13] (Proposition 4.15 et Corollaire 4.19) :

Proposition 6.2.51. *Soit $f_1, \dots, f_s \in \mathcal{A}^s$ des polynômes homogènes. Soit $I = \langle f_1, \dots, f_s \rangle$, alors génériquement, si $x^\alpha \in LT_{\text{grevlex}}(I)$ et $x_n | x^\alpha$, on a pour tout $k \in \llbracket 1, n \rrbracket$ $x_k \frac{x^\alpha}{x_n} \in LT_{\text{grevlex}}(I)$. De plus, sur un corps infini, un changement de variable linéaire générique nous ramène à ce cas.*

Cette proposition a la conséquence suivante (Théorème 4.16 de [Huo13]) :

Proposition 6.2.52. *Génériquement, la matrice de multiplication T_n pour l'ordre grevlex d'un idéal I peut être obtenue sans opération arithmétique à partir de la base de Gröbner réduite G de I pour grevlex.*

En effet, dans ces conditions, les monômes de la forme $x_n \varepsilon$ avec $\varepsilon \in \mathcal{B}_{\text{grevlex}}(I)$ sont soit dans $\mathcal{B}_{\text{grevlex}}(I)$, soit dans $LM(G)$.

Maintenant, avec les Propositions 6.2.52 et 6.2.50, nous pouvons proposer la version suivante de FGLM, adaptée au calcul d'une base de Gröbner pour l'ordre lexicographique d'un idéal en position générale à partir d'une base de Gröbner pour l'ordre grevlex s'il satisfait la Proposition 6.2.51.

Algorithme 6.2.53 : Algorithme FGLM pour un idéal en position générale à partir de grevlex

entrée : Une base de Gröbner réduite G_1 pour l'ordre grevlex d'un idéal $I \subset \mathcal{A}$ de dimension zéro et de degré δ , $B_{\leq} = (1 = \epsilon_1 \leq \epsilon_2 \leq \dots \leq \epsilon_\delta)$ la base canonique de \mathcal{A}/I pour \leq .
On exige que I satisfasse la Proposition 6.2.51 I est en position générale.

sortie : Une base de Gröbner G_2 de I pour \leq_{lex} .

début

Calculer la matrice de multiplication T_n pour I et grevlex à partir de G (Prop 6.2.52) ;
 $G_2 := \emptyset$;
pour i de 1 à $n-1$ **faire**
 Grâce à la Prop. 6.2.50 calculer $\mathbf{y}[i] := T_i 1$ à partir de G ;
 $\mathbf{z}[0] := 1$;
pour i de 1 à δ **faire**
 Calculer $\mathbf{z}[i] = T_n \mathbf{z}[i-1]$;
 $M := \text{Mat}_{B_{\leq}}(\mathbf{z}[0], \dots, \mathbf{z}[\delta-1])$;
pour i de 1 à $n-1$ **faire**
 Trouver U tel que $\mathbf{y}[i] = -M \cdot U$;
 $h_i(T) := \sum_{j=1}^{\delta-1} U[j] T^j$;
 Trouver U tel que $\mathbf{z}[\delta] = -M \cdot U$;
 $h_n(T) := T^\delta + \sum_{j=1}^{\delta-1} U[j] T^j$;
 Retourner $x_1 - h_1(x_n), \dots, x_{n-1} - h_{n-1}(x_n), h_n(x_n)$;

Nous avons alors le résultat suivant :

Proposition 6.2.54. *Ayant en entrée une base de Gröbner réduite G_1 d'un idéal de dimension zéro et de degré δ $I \subset \mathcal{A}$ pour grevlex, en supposant que I est en position générale et satisfait le résultat de la Proposition 6.2.50, alors l'Algorithme 6.2.53 termine et renvoie bien une base de Gröbner G_2 de I pour \leq_{lex} . Ces hypothèses sont vérifiées génériquement, à changement de variable générique près. Le temps de calcul est en $O(\delta^3) + O(n\delta^2)$.*

Démonstration. Pour ce qui est de la correction et de la terminaison, c'est une conséquence claire des Propositions 6.2.49, 6.2.50 et 6.2.52. Maintenant, le temps de calcul pour calculer T_n est en $O(1)$ opérations arithmétiques, et nous n'avons plus besoin des T_i . La complexité s'en déduit. \square

Version plus avancées Des versions plus efficaces ou plus générales de ces algorithmes sont disponibles. Outre l'extension à des hypothèses moins restrictives, nous pouvons citer deux types d'améliorations :

6. Algorithmes classiques pour le calcul de bases de Gröbner

- Dans [FM11, FM13, Mou13] est étudié le cas où les matrices opérées au cours de l'algorithme sont creuses, ainsi que l'application d'algorithmes variantes de celui de Berlekamp-Massey pour le calcul des h_i ;
- Dans [FGHR13, FGHR14, Huo13] est étudié de plus le fait que ces algorithmes de Berlekamp-Massey permettent de se ramener à la résolution de systèmes de Hankel. Or, il existe des algorithmes rapides adaptés à ces systèmes, permettant de résoudre le problème du changement d'ordre monomial en un temps sous-cubique.

Nous n'étudierons pas plus en détail ces variantes plus avancées car leur application au cas de la précision finie (lorsque cela est possible) n'apparaît pas aisée. Ceci est dû au fait, notamment, qu'elles utilisent beaucoup de conditions d'arrêt reposant sur des tests à zéro, en particulier dans les algorithmes de type Berlekamp-Massey. L'auteur espère que de futurs travaux s'intéresseront à cette question.

7. Algorithme F5-Matriciel et stabilité

"The world ends with you. If you
want to enjoy life, expand your world.
You gotta push your horizons out as
far as they'll go."

Hanekoma, *The World Ends With
You*

"It is well known that a vital
ingredient of success is not knowing
that what you're attempting can't be
done."

Terry Pratchett, *Equal Rites*

Ce chapitre est une version étendue d'un article publié lors de la conférence ISSAC 2014 [Vac14]. Il s'intéresse au problème du calcul direct d'une base de Gröbner par un algorithme F5-Matriciel, éventuellement modifié.

Nous présentons globalement les résultats de cette partie en Section 7.1. Ensuite, la Section 7.2 introduit avec plus de détails les conditions, ouvertes, que nous énonçons comme suffisantes pour calculer une base de Gröbner (approchée) d'un système polynomial dont les coefficients sont pris dans un CDVF à précision finie, ainsi qu'une adaptation d'un algorithme F5-Matriciel pour effectuer ce calcul. La Section 7.3 présente les conséquences des résultats de la Section précédente du point de vue de la topologie (continuité, différentiabilité) et de l'optimalité des hypothèses. La Section 7.4 présente une illustration numériques des résultats précédents, en s'intéressant notamment au comportement effectif de la précision. Parmi les conséquences des résultats de la Section 7.2, la Section 7.5 présente un algorithme de remontée modulaire permettant, lorsqu'on augmente la précision sur les coefficients des polynômes en entrée, d'augmenter efficacement la précision sur les coefficients d'une base de Gröbner de ces polynômes. Enfin, la Section 7.6 traite de l'extension des résultats précédents aux calculs de bases de Gröbner d'idéaux non-homogènes et aux calculs de bases de Gröbner réduites.

7.1. Présentation des résultats de cette partie

Soit K un CDVF à précision finie, par exemple $K = \mathbb{Q}_p$ ou $\mathbb{F}_p((t))$, et soit $R = O_K$, ce qui correspond dans les exemples précédents à $R = \mathbb{Z}_p$ ou $\mathbb{F}_p[[t]]$ respectivement. Notons $A = K[X_1, \dots, X_n]$ et $B = R[X_1, \dots, X_n]$. Soit ω un ordre monomial sur $K[X_1, \dots, X_n]$ et soit $f = (f_1, \dots, f_s) \in B^s$ des polynômes homogènes satisfaisant les deux hypothèses suivantes :

- **H1** : (f_1, \dots, f_s) est une suite régulière.
- **H2** : les idéaux $\langle f_1, \dots, f_i \rangle$ sont des idéaux faiblement- ω (voir la Définition 7.2.4).

Ces hypothèses assurent la régularité du calcul d'une base de Gröbner : sur un voisinage de f où les polynômes satisfont **H1** et **H2**, l'application envoyant une famille finie de polynômes sur sa base de Gröbner réduite pour ω est différentiable (et continue), et nous pouvons calculer explicitement la différentielle grâce au Théorème 7.3.2. Ainsi, au voisinage d'un tel f , il est possible de travailler avec des approximations. Au contraire, si l'on relâche l'une des hypothèses **H1** ou **H2**, la continuité n'est plus garantie (voir la Section 7.3), ce qui signifie que le calcul peut ne plus être possible à précision finie. De manière plus précise, nous rendons explicite une précision

$$prec_{MF5}(\{f_1, \dots, f_s\}, D, \omega),$$

7. Algorithme F5-Matriciel et stabilité

donnée par les mineurs des matrices de Macaulay définies par f , tel que les approximations de f à précision au moins $prec_{MF5}$ déterminent bien des bases de Gröbner approchées, compatibles avec la précision et sans ambiguïté sur les termes de tête. Nous fournissons en Définition 7.2.1 une notion adaptée de base de Gröbner approchée lorsqu'on travaille à précision finie. Nous définissons de même les D -bases de Gröbner approchées. Pour calculer de telles D -bases de Gröbner, nous définissons avec l'Algorithme 7.2.7 un algorithme F5-Matriciel faible, et nous avons le résultat suivant :

Théorème 7.1.1. *Soit $(f_1, \dots, f_s) \in K[X_1, \dots, X_n]^s$ des polynômes homogènes satisfaisant **H1** et **H2**. Soit (f'_1, \dots, f'_s) des approximations des f_i à précision $O(p^m)$ sur les coefficients. Alors, si m est assez grand, une D -base de Gröbner approchée de (f'_1, \dots, f'_s) par rapport à ω est bien définie. Elle peut être calculée par l'algorithme F5-Matriciel faible.*

De plus, si les f_i sont dans $R[X_1, \dots, X_n]$, alors $m \geq prec_{MF5}$ est suffisant, et la perte de précision est majorée par $prec_{MF5}$.

La complexité est en $O\left(sD\binom{n+D-1}{D}^3\right)$ opérations dans R à précision m , lorsque $D \rightarrow +\infty$.

Nous remarquons alors que la conjecture de Moreno-Socias implique que les suites de polynômes satisfaisant **H1** et **H2** pour l'ordre grevlex sont génériques.

Nous expliquons en Section 7.4 pourquoi la borne $prec_{MF5}$ n'est pas optimale, en l'accompagnant d'exemples numériques.

Si l'on préfère la précision au temps de calcul, nous montrons avec le Théorème 7.2.17 que, sous les hypothèses **H1** et **H2** et les f_i dans $R[X_1, \dots, X_n]$, nous pouvons abandonner le critère F5 afin d'obtenir une majoration plus petite de la précision pour calculer une base de Gröbner approchée : $prec_{Mac}$, voir la Définition 7.2.16. La complexité est alors en $O\left(s^2D\binom{n+D-1}{D}^3\right)$ opérations sur R à précision m , lorsque $D \rightarrow +\infty$.

En outre, les hypothèses **H1** et **H2** permettent de remonter des bases de Gröbner : étant donné G une base de Gröbner approchée de $\langle F \rangle$ et des m, k et M tels que $(G + O(p^k)) = (F + O(p^m)) \cdot (M + O(p^m))$, on peut alors calculer en $O\left((s + \#G) \binom{n+D-1}{D}^2\right)$ opérations à précision $m + l$ une base de Gröbner approchée de $F + O(p^{m+l})$. $\#G$ est ici utilisé pour le cardinal de G . Ceci implique que pour le calcul de bases de Gröbner réduites de G $F = (f_1, \dots, f_s) \in \mathbb{Z}[X_1, \dots, X_n]$ satisfaisant **H1** et **H2**, il est suffisant d'effectuer d'abord le calcul d'une base de Gröbner approchée avec une précision en entrée m suffisamment grande puis de la remonter dans $\mathbb{Q}[X_1, \dots, X_n]$. La complexité totale est alors en $O\left(s^2D\binom{n+D-1}{D}^3\right)$ opérations dans \mathbb{Z}_p à précision m et $O\left((s + \#G) \binom{n+D-1}{D}^2\right)$ opérations dans \mathbb{Q} . En d'autres termes, le coût de l'algèbre linéaire est porté par le calcul à précision fini sur \mathbb{Z}_p .

Enfin, même si les résultats précédents étaient présentés pour des polynômes en entrée homogènes, ils peuvent s'étendre au cas général en supposant cette fois les hypothèses **H1** et **H2** vérifiées par leurs composantes homogènes de plus haut degré.

7.2. Algorithme F5-Matriciel et calcul de bases de Gröbner à précision finie

7.2.1. Problèmes de précision

Nous nous intéressons maintenant à ce qu'il advient lors du calcul de bases de Gröbner par l'algorithme F5-Matriciel lorsque les entrées sont connues seulement à précision finie. À cette fin, nous donnons une définition de ce que nous appelons une base de Gröbner approchée :

Définition 7.2.1. Soit $f_i + \sum_{|u|=d_i} O(\pi^{n_{u,i}})X^u$, $1 \leq i \leq s$, des approximations de polynômes homogènes de A , avec les $n_{u,i}$ dans $\mathbb{Z}_{\geq 0} \cup \{+\infty\}$. Une base de Gröbner approchée, par rapport à un ordre monomial ω , de l'idéal engendré par les $f_i + \sum_{|u|=d_i} O(\pi^{n_{u,i}})X^u$ est une suite finie $(g_i + \sum_{|u|=|g_i|} O(\pi^{m_{u,i}})X^u)$, $m_{u,i} \in \mathbb{Z}_{\geq 0} \cup \{+\infty\}$, d'approximations de polynômes tels que : pour tout $a_{u,i} \in \pi^{n_{u,i}}R$, il existe des $b_{u,i} \in \pi^{m_{u,i}}R$ tels que les $g_i + \sum_{|u|=|g_i|} b_{u,i}X^u$ forment une base de Gröbner, par rapport à ω , de l'idéal engendré par les $f_i + \sum_{|u|=d_i} a_{u,i}X^u$'s. De plus, on demande que si X^u est un monôme de degré $|g_i|$ tel que $X^u >_{\omega} LM(g_i)$, alors $m_{u,i} = +\infty$ (et le coefficient de X^u

pour g_i est zéro). En d'autres mots, on demande que les termes de tête des g_i soient indépendants de l'approximation faite sur les f_i .

Comme vu dans 6.2.11, si les polynômes en entrée forment une suite régulière, alors toutes les matrices considérées dans l'algorithme F5-Matriciel sont injectives. Cependant, cela n'est pas suffisant pour pouvoir garantir quels sont les termes de tête, et ainsi que l'on a bien une base de Gröbner approchée.

Par exemple, la matrice injective,

$$\begin{bmatrix} 1 + O(\pi^{10}) & 1 + O(\pi^{10}) & 1 + O(\pi^{10}) & 0 \\ 1 + O(\pi^{10}) & 1 + O(\pi^{10}) & 1 + O(\pi^{10}) & 1 + O(\pi^{10}) \end{bmatrix},$$

devient, après la première étape de calcul d'une forme échelonnée en lignes :

$$\begin{bmatrix} 1 + O(\pi^{10}) & 0 & 1 + O(\pi^{10}) & 0 \\ 0 & O(\pi^{10}) & O(\pi^{10}) & 1 + O(\pi^{10}) \end{bmatrix}.$$

Cependant, il n'est pas possible, avec seulement des opérations sur les lignes, de déterminer si le coefficient d'indice $(2, 2)$ est le premier coefficient non-nul de la seconde ligne, ou s'il s'agit de celui d'indice $(2, 3)$ ou $(2, 4)$. Ainsi, nous ne pouvons pas savoir quel serait le terme de tête du polynôme qui correspondrait à cette seconde ligne si cette matrice était une matrice de Macaulay.

En conséquence, les (f_1, \dots, f_s) tels que l'algorithme F5-Matriciel puisse donner une réponse satisfaisante doivent avoir une forme particulière : lorsque l'on calcule une forme échelonnée en lignes d'une matrice de Macaulay, aucune colonne sans pivot ne doit être rencontrée. Si ω est notre ordre monomial, un idéal $\langle f_1, \dots, f_s \rangle$ tel que toutes les matrices $Mac_d(f_1, \dots, f_j)$ satisfont cette propriété est appelé un ω -idéal.

Les ω -idéaux ont été étudiés en détail dans le cadre de l'étude des idéaux de tête génériques (*Generic Initial Ideal*, ou *gin* en anglais). Une excellente introduction à ce domaine est la partie 15.9 de [Eis95]. Nous rappelons la définition d'idéal de tête générique d'un idéal. Elle repose sur la proposition suivante :

Proposition 7.2.2. *Soit I un idéal homogène de A et ω un ordre monomial sur A . Nous supposons que k est infini. Alors il existe U , ouvert de Zariski non-vide de $GL_n(k)$ et I_0 idéal monomial de A tel que pour tout $g \in U$, si l'on note J l'idéal obtenu par changement de variable linéaire avec la matrice g , on a $LM(J) = I_0$.*

Définition 7.2.3. Dans le contexte précédent, nous notons $I_0 = gin(I)$, et nous l'appelons l'idéal de tête générique de I (pour ω).

Pour ce qui est des liens entre idéaux de tête génériques et ω -idéaux, on peut citer [CS05], où Conca et Sidman prouvent que l'idéal de tête générique d'un ensemble de points génériques dans \mathbb{P}^r est un ω -idéal.

Cependant, ce n'est pas le cas génériquement pour n'importe quel choix de degrés pour les polynômes homogènes en entrée. Par exemple, Pardue a montré dans [Par10] que l'idéal engendré par 6 quadriques en 6 variables n'est pas un idéal grevlex.

Heureusement, on peut étudier une condition légèrement plus faible, celle d'idéal faiblement- ω .

Définition 7.2.4. Soit I un idéal de A , et ω un ordre monomial sur A . Alors I est appelé un idéal faiblement- ω si, pour tout monôme de tête x^α de la base de Gröbner réduite de I , par rapport à ω , pour tout x^β tel que $|\alpha| = |\beta|$ et $x^\beta >_\omega x^\alpha$, x^β appartient à $LM_\omega(I)$.

Remarque 7.2.5. Tout ω -idéal est bien sûr un idéal faiblement- ω . La réciproque est fausse. Par exemple, $I = \langle x, y^3 \rangle \in \mathcal{K}[x, y, z]$ est un idéal faiblement-grevlex qui n'est pas un grevlex-idéal. En effet, sa base de Gröbner réduite pour grevlex ($x > y > z$) est (x, y^3) . Nous vérifions aisément que tous les monômes de degré 3 plus grands que y^3 sont bien dans $LM(I)$. Par contre, en degré 2, nous avons $xy > y^2 > xz$ avec $xy, xz \in LM(I)$ mais pas y^2 .

Moreno-Socias a conjecturé que la propriété d'être faiblement-grevlex était générique au sens suivant :

Conjecture 1 (Moreno-Socias). *Si k est un corps infini, $s \in \mathbb{N}$, $d_1, \dots, d_s \in \mathbb{N}$, alors il existe un ouvert de Zariski non-vide U de $A_{d_1} \times \dots \times A_{d_s}$ tel que pour tout $(f_1, \dots, f_s) \in U$, $I = (f_1, \dots, f_s)$ est un idéal faiblement-grevlex.*

En conséquence, si la conjecture de Moreno-Socias est vérifiée, les suites satisfaisant les propriétés **H1** et **H2** sont génériques. Nous renvoyons à la thèse de doctorat de Moreno-Socias [MS91] ou à l'article [Par10] de Pardue pour une introduction à cette conjecture.

Remarque 7.2.6. Le choix de grevlex est important : comme on peut le voir dans [Par10], si l'on prend 3 quadriques (f_1, f_2, f_3) dans $\mathbb{Q}[X_1, \dots, X_6]$, alors génériquement, l'idéal I qu'elles engendrent n'est ni lex, ni faiblement-lex ! En effet, pour lex donné par $X_1 > \dots > X_6$, les termes de tête de I en degré 2 sont, génériquement, X_1^2 , X_1X_2 et X_1X_3 . Cependant, en degré 3, on a génériquement $X_2^3 \in LM(I)$ et $X_1X_6^2 \notin LT(I)$ alors que $X_1X_6^2 > X_2^3$, et X_2^3 n'est le multiple d'aucun des termes de tête de I en degré 2, X_1^2 , X_1X_2 et X_1X_3 . Donc, X_2^3 est un générateur minimal de $LM(I)$, mais $X_1X_6^2 > X_2^3$ et $X_1X_6^2 \notin LT(I)$. Ainsi, l'idéal engendré par 3 quadriques génériques en 6 variables n'est ni lex, ni faiblement-lex.

7.2.2. L'Algorithme F5-Matriciel-Faible

Nous fournissons maintenant, avec l'Algorithme 7.2.7 un algorithme que nous appellerons F5-Matriciel-Faible. L'objectif sera de montrer que si (f_1, \dots, f_s) est une suite de polynômes homogènes de A satisfaisant **H1** et **H2**, et si les f_i sont connus avec une précision suffisamment grande $O(\pi^k)$ sur leurs coefficients, alors l'algorithme F5-Matriciel-Faible calcule une D -base de Gröbner approchée de $\langle f_1, \dots, f_s \rangle$.

Algorithme 7.2.7 : L'algorithme F5-Matriciel-Faible

entrée : $F = (f_1, \dots, f_s) \in R[X_1, \dots, X_n]^s$, polynômes homogènes de degré $d_1 \leq \dots \leq d_s$, et $D \in \mathbb{N}$, un ordre monomial ω .
sortie : $(g_1, \dots, g_k) \in A^k$, une D -base de Gröbner de $\langle F \rangle$, ou **Erreur** si F ne satisfait pas **H1**, **H2** ou que la précision n'est pas suffisante.

début

```

     $G \leftarrow \{\}$  ;
    pour  $d \in \llbracket 0, D \rrbracket$  faire
         $\widetilde{\mathcal{M}}_{d,0} := \emptyset$  ;
        pour  $i \in \llbracket 1, s \rrbracket$  faire
             $\mathcal{M}_{d,i} := \mathcal{M}_{d,i-1}$  ;
            pour  $\alpha$  tel que  $|\alpha| + d_i = d$  faire
                si  $x^\alpha$  n'est pas le terme de tête d'une ligne de  $\widetilde{\mathcal{M}}_{d-d_i,i-1}$  alors
                    Ajouter  $x^\alpha f_i$  à  $\mathcal{M}_{d,i}$  ;
            Calculer  $\widetilde{\mathcal{M}}_{d,i}$ , la forme échelonnée de  $\mathcal{M}_{d,i}$ , jusqu'à la première colonne sans pivot non-nul ;
            Remplacer les lignes restantes de  $\widetilde{\mathcal{M}}_{d,i}$  par des multiples de lignes de  $\widetilde{\mathcal{M}}_{d-1,i}$ , afin d'obtenir une matrice sous forme échelonnée en lignes  $\widetilde{\mathcal{M}}_{d,i}$  ;
            si  $\widetilde{\mathcal{M}}_{d,i}$  n'a pas pu être complétée alors
                Retourner "Erreur, les idéaux ne sont pas faiblement- $\omega$ , la suite n'est pas régulière, ou la précision est insuffisante" ;
            sinon
                Ajouter à  $G$  les polynômes des lignes de  $\widetilde{\mathcal{M}}_{d,i}$  avec un nouveau monôme de tête ;
    Retourner  $G$  ;
    
```

Remarque 7.2.8. Au début de la second boucle *pour*, l'algorithme F5-Matriciel classique utilise $\mathcal{M}_{d,i} := \widetilde{\mathcal{M}}_{d,i-1}$ à la place de $\mathcal{M}_{d,i} := \mathcal{M}_{d,i-1}$. La première possibilité donne un algorithme plus

rapide ($\widetilde{\mathcal{M}}_{d,i-1}$ est déjà sous-forme échelonnée en ligne), mais nous avons choisi la seconde car l'analyse de la précision y est plus aisée, et plus efficace.

Remarque 7.2.9. Au lieu d'ajouter à G tous les polynômes des lignes de $\widetilde{\mathcal{M}}_{d,i}$ ayant un nouveau terme de tête, il serait suffisant d'ajouter seulement ceux des lignes dont le terme de tête n'est pas un multiple du terme de tête d'un polynôme de G , ainsi, on obtiendrait directement une (D) -base de Gröbner minimale en sortie.

Correction

Nous prouvons ici que formellement (*i.e.* sans considération de précision) l'algorithme F5-Matriciel-Faible calcule bien des D -bases de Gröbner.

Proposition 7.2.10. *Soit $(f_1, \dots, f_s) \in B^s = R[X_1, \dots, X_n]^s$ une suite de polynômes homogènes satisfaisant **H1** et **H2**. Alors, pour tout $D \in \mathbb{Z}_{\geq 0}$, le résultat de F5-Matriciel-Faible($(f_1, \dots, f_s), D$) est une D -base de Gröbner de l'idéal I engendré par (f_1, \dots, f_s) . Si (f_1, \dots, f_s) ne satisfait pas **H1** ou **H2**, une erreur est renvoyée.*

Démonstration. Soit $(f_1, \dots, f_s) \in B^s$, homogènes de degré $d_1 \leq \dots \leq d_s$ et satisfaisant **H1** et **H2**. Soit $\mathcal{M}_{d,i}$ la matrice construite avec le critère F5 au début de la deuxième boucle **pour** dans l'Algorithme 7.2.7, et $\widetilde{\mathcal{M}}_{d,i}$ le résultat à la fin de cette même boucle, obtenue à partir de $\mathcal{M}_{d,i}$ par calcul de forme échelonnée et complétion avec $\widetilde{\mathcal{M}}_{d-1,i}$.

Soit $\mathcal{P}(d, i)$ la proposition : $\mathcal{M}_{d,i} = \overline{\text{Mac}_d(f_1, \dots, f_i)}$, $\widetilde{\mathcal{M}}_{d,i}$ sous forme échelonnée (à permutation des lignes près), aucune **Erreur** n'a été levée, et $\text{Im}(\widetilde{\mathcal{M}}_{d,i}) = \text{Im}(\text{Mac}_d(f_1, \dots, f_i))$. Nous prouvons par récurrence sur d et i que pour tout $d \in [0, D]$ et $i \in [1, s]$, $\mathcal{P}(d, i)$ est vraie.

Tout d'abord, $\mathcal{P}(d, i)$ est clairement vraie pour $d < d_1$ puisque ces matrices sont vides.

Maintenant, soit $d \in [d_1, D]$ tel que pour tout $0 \leq \delta \leq d$ et $i \in [1, s]$, $\mathcal{P}(\delta, i)$ est vraie. Nous prouvons $\mathcal{P}(d, i)$ pour tout $i \in [1, s]$.

Cela est clair pour $i = 1$ puisque l'idéal engendré par f_1 est monogène. Soit $i \in [1, s]$ tel que pour tout $j \in [1, i-1]$, $\mathcal{P}(\delta, i)$ est vraie.

Alors, par hypothèse de récurrence $\mathcal{M}_{d,i-1} = \overline{\text{Mac}_d(f_1, \dots, f_{i-1})}$, $\widetilde{\mathcal{M}}_{d-d_i, i-1}$ est sous forme échelonnée (à permutation des lignes près) et $\text{Im}(\widetilde{\mathcal{M}}_{d-d_i, i-1}) = \text{Im}(\text{Mac}_d(f_1, \dots, f_{i-1}))$. Par le critère F5 (Proposition 6.2.9), nous avons alors $\mathcal{M}_{d,i} = \overline{\text{Mac}_d(f_1, \dots, f_i)}$. Ensuite, suivant l'algorithme, $\widetilde{\mathcal{M}}_{d,i}$ reçoit la forme échelonnée de $\mathcal{M}_{d,i}$ jusqu'à la première colonne sans pivot non-nul.

Nous prouvons maintenant, que le procédé de complétion est effectué sans erreur. Soit x^{α_u} , pour u de 1 à $\binom{n+d-1}{n-1}$, les monômes de degré d , triés par ordre décroissant selon ω , et soit l l'indice de la première colonne sans pivot non-nul trouvé durant le calcul de la forme échelonnée en ligne par pivot de Gauss de $\mathcal{M}_{d,i}$. Soit r_i , pour i de 1 à $l-1$, les l polynômes correspondant aux lignes de $\text{Mac}_d(f_1, \dots, f_s)$ avec monômes de tête x^{α_i} . Leurs monômes de tête sont dans x^{α_u} , avec $u \geq l$.

Soit (g_1, \dots, g_r) la base de Gröbner réduite de I par rapport à ω . Alors, puisqu'il n'y a pas de pivot non nul sur la colonne d'indice l , x^{α_l} n'est pas un monôme de $LM(I)$. Par définition d'un idéal faiblement- ω (hypothèse **H2**), ceci implique que si $x^{\alpha_u} \in LM(I)$ pour un certain $u \geq l$, alors x^{α_u} n'est pas l'un des $LM(g_i)$. Cela veut dire que $x^{\alpha_u} \in LM(I)$ avec $u \geq l$ est un multiple non-trivial des $LM(g_i)$. Ainsi x^{α_u} est un multiple d'un monôme de $LM(I \cap A_{d-1})$. En conséquence, comme $\text{Im}(\widetilde{\mathcal{M}}_{d-1,i}) = I \cap A_{d-1}$, et $\widetilde{\mathcal{M}}_{d-1,i}$ est sous forme échelonnée (à permutation près), alors pour tout $u \geq l$ tel que $x^{\alpha_u} \in LM(I)$, il existe un polynôme P_u correspondant à une ligne de $\widetilde{\mathcal{M}}_{d-1,i}$ et $k_u \in [1, n]$ tel que $LM(X_{k_u} P_u) = x^{\alpha_u}$. Avec le critère F5 et l'hypothèse **H1**, $\mathcal{M}_{d,i} = \overline{\text{Mac}_d(f_1, \dots, f_i)}$, $\mathcal{M}_{d,i}$ est injective et le nombre de ses lignes, m , est exactement le nombre de monômes dans $LM(I \cap A_d)$. Ceci implique que le procédé de complétion se passe sans lever d'erreur.

Soit (t_1, \dots, t_m) les lignes de $\widetilde{\mathcal{M}}_{d,i}$ obtenues comme multiples des lignes de $\widetilde{\mathcal{M}}_{d-1,i}$. Alors les polynômes correspondant aux lignes $(r_1, \dots, r_{l-1}, t_1, \dots, t_m)$ ont des termes de têtes distincts, et ainsi, $\widetilde{\mathcal{M}}_{d,i}$ est sous forme échelonnée (à permutation des lignes près). Finalement, $\text{Im}(\widetilde{\mathcal{M}}_{d,i}) \subset$

7. Algorithme F5-Matriciel et stabilité

$I \cap R_d = \text{Im}(\text{Mac}_d(f_1, \dots, f_i))$ et ont tous les deux la même dimension m sur K . Ainsi, $\text{Im}(\widetilde{\mathcal{M}}_{d,i}) = \text{Im}(\text{Mac}_d(f_1, \dots, f_i))$.

$\mathcal{P}(d, i)$ est donc prouvée. Par récurrence, elle est donc vraie pour tout $d \in \llbracket 0, D \rrbracket$ et $i \in \llbracket 1, s \rrbracket$. En conséquence, la sortie de l'Algorithme 7.2.7 est bien une D -base de Gröbner de (f_1, \dots, f_s) .

Maintenant, si (f_1, \dots, f_s) n'est pas régulière ou il existe un i tel que (f_1, \dots, f_i) n'est pas faiblement- ω . Dans le premier cas, cela veut dire que certaines lignes des $\text{Mac}_d(f_1, \dots, f_i)$ se réduisent à zéro (voir par exemple [Bar04]). Ainsi, le calcul de la forme échelonnée rencontrera une colonne sans pivot, et la complétion de $\mathcal{M}_{d,i}$ en une base échelonnée ne sera pas possible, levant une erreur. L'étude du deuxième cas est similaire. \square

Remarque 7.2.11. Comme vu dans la preuve, le procédé de complétion est seulement là pour pouvoir certifier qu'il ne manque pas de termes de tête et qu'ainsi, on obtient bien une D -base de Gröbner en sortie. Il ne fournit pas de nouveau polynôme pour la base de Gröbner en cours de calcul. Si K avait été un corps exact, alors avec l'hypothèse **H2**, arrêter le calcul de la forme échelonnée en ligne au moment où la première colonne sans pivot est trouvée est suffisant pour obtenir une base de Gröbner en sortie. Le critère de Buchberger est alors suffisant pour certifier qu'il s'agit bien d'une base de Gröbner.

Terminaison

Comme nous nous restreignons à calculer des formes échelonnées de matrices de Macaulay jusqu'au degré D , il n'y a pas de problème de terminaison.

Cependant, si l'on veut obtenir en sortie une base de Gröbner, et non juste une D -base de Gröbner, nous pouvons utiliser le résultat suivant : (voir [BFS14], [Giu84], [Laz83])

Proposition 7.2.12. *Soit (f_1, \dots, f_n) une suite régulière de polynômes homogènes de A . Alors, après un changement de variables générique, le plus haut degré d'un élément de la base de Gröbner réduite de $\langle f_1, \dots, f_n \rangle$ pour l'ordre grevlex est majorée par la borne de Macaulay : $\sum_{i=1}^n (|f_i| - 1) + 1$.*

Précision

Nous pouvons maintenant prouver le Théorème 7.1.1. Soit (f_1, \dots, f_s) une suite de polynômes homogènes de B satisfaisant **H1** et **H2**. Nous définissons pour cela $\Delta_{d,i}$, qui sera par la Proposition 1.2.7 suffisante pour calculer $\widetilde{\mathcal{M}}_{d,i}$ à partir de $\mathcal{M}_{d,i}$.

Définition 7.2.13. Soit $l_{d,i}$ le maximum des $l \in \mathbb{Z}_{\geq 0}$ tels que les l premières colonnes de $\text{Mac}_d(f_1, \dots, f_i) = \mathcal{M}_{d,i}$ sont linéairement indépendantes. Nous définissons

$$\Delta_{d,i} = \min(\text{val}(\{\text{mineur sur les } l_{d,i}\text{-premières colonnes de } \mathcal{M}_{d,i}\})).$$

Nous pouvons maintenant définir prec_{F5M} .

Définition 7.2.14. Nous définissons la précision F5-Matricielle de $(f_1, \dots, f_s) \in B^s$, homogènes, par rapport à ω et D comme :

$$\text{prec}_{F5M}((f_1, \dots, f_s), D, \omega) = \max_{d \leq D, 1 \leq i \leq s} (\Delta_{d,i}).$$

Avec la Proposition 1.2.7 et la Proposition 7.2.10, prec_{F5M} est une borne supérieure sur la précision suffisante pour calculer les $\widetilde{\mathcal{M}}_{d,i}$ pour d jusqu'à D . En effet, il est suffisant de calculer les formes échelonnées en lignes des matrices $\mathcal{M}_{d,i}$ jusqu'à la colonne $l_{d,i}$ et ensuite de compléter ces matrices avec des multiples de lignes de $\widetilde{\mathcal{M}}_{d-1,i}$. De cette manière, qu'une ligne de $\widetilde{\mathcal{M}}_{d,i}$ proviennent de l'échelonnement partiel ou d'un multiple d'une ligne de $\widetilde{\mathcal{M}}_{d-1,i}$, son terme de tête est défini sans ambiguïté. Le fait que le procédé de complétion soit effectué avec succès implique nous pouvons certifier que l'on a bien obtenu une base échelonnée de $\text{Im}(\mathcal{M}_{d,i})$.

En conséquence, $\text{prec}_{F5M}((f_1, \dots, f_s), D, \omega)$ est suffisant pour calculer des D -bases de Gröbner approchées par l'Algorithme F5-Matriciel faible.

Pour conclure cette preuve, nous remarquons que, afin de faciliter notre étude, nous avons supposé que les polynômes en entrée (f_1, \dots, f_i) , sont dans B . Cependant, si (f_1, \dots, f_i) , sont dans A , nous

pouvons toujours nous ramener à avoir des f_i dans B en les multipliant par des $\pi^{l_i} \in O_K$ (pour de bons l_i) et toujours engendrer le même idéal. Cela n'affecte pas **H1** et **H2**, et nous pouvons ainsi toujours calculer une base de Gröbner approchée dès que la précision est suffisante. Seule notre borne précise $prec_{F5M}$ n'est plus disponible.

Complexité

La complexité du calcul d'une forme échelonnée pour une matrice injective de n_{lignes} lignes et n_{col} colonnes est en $O(n_{rows}^2 \times n_{cols})$ opérations dans K . Nous en déduisons la complexité suivante pour l'Algorithme F5-Matriciel-Faible : $O\left(sD\binom{n+D-1}{D}^3\right)$ opérations dans R , à précision donnée $O(\pi^m)$, pour $D \rightarrow +\infty$. Comparé au cas classique (voir 6.2.13) la complexité subit essentiellement l'ajout d'un facteur s . Ceci vient du fait que nous avons choisi, pour des raisons de stabilité, de calculer entièrement la forme échelonnée pour chaque nouveau $\mathcal{M}_{d,i}$. En d'autres termes, on ne prend pas en compte le fait que lors de la construction de la matrice $\mathcal{M}_{d,i}$, $\widetilde{\mathcal{M}_{d,i-1}}$ est déjà sous forme échelonnée.

Un exemple

Nous reprenons l'exemple 6.2.15, à précision finie :

Exemple 7.2.15. Nous appliquons l'Algorithme F5-Matriciel Faible 7.2.7 sur $F = (f_1, f_2, f_3) \in \mathbb{Q}_2[x, y, z]$ pour grevlex avec les f_i connus à précision $O(2^{10})$: $f_1 = (2 + O(2^{10}))x + (1 + O(2^{10}))z$, $f_2 = (1 + O(2^{10}))x^2 + (1 + O(2^{10}))y^2 - (2 + O(2^{10}))z^2$ et $f_3 = (4 + O(2^{10}))y^2 + (1 + O(2^{10}))yz + (8 + O(2^{10}))z^2$. Nous prenons $D = 3$, la borne de Macaulay.

Alors, le calcul se passe exactement comme dans l'exemple 6.2.15 avec l'Algorithme 6.2.12 à la différence suivante : les matrices $\mathcal{M}_{2,1}$ et $\mathcal{M}_{3,1}$ demandent le remplacement de certaines de leurs lignes puisqu'elles contiennent des colonnes sans pivots.

Nous obtenons *in fine* : $\widetilde{\mathcal{M}_{1,3}}$ qui est

$$f_1 \begin{array}{c} x \quad y \quad z \\ \left| \begin{array}{ccc} 2 + O(2^{10}) & 0 & 1 + O(2^{10}) \end{array} \right|, \end{array}$$

$$\widetilde{\mathcal{M}_{2,3}} = \begin{array}{c} \begin{array}{c} x^2 \quad xy \quad y^2 \quad xz \quad yz \quad z^2 \\ \begin{array}{c} zf_1 \\ yf_1 \\ xf_1 \\ f_2 \\ f_3 \end{array} \end{array} \left| \begin{array}{ccccc} & & & 2 & & 1 \\ & & & & & 1 \\ 0 & & -2 & 1 & & 4 \\ 1 & & 1 & & & -2 \\ & & & & 1 & 15 \end{array} \right| \begin{array}{c} +O(2^{10}) \\ +O(2^{10}) \\ +O(2^{10}) \\ +O(2^{10}) \\ +O(2^8) \end{array} \end{array}$$

7. Algorithme F5-Matriciel et stabilité

et

$$\widetilde{\mathcal{M}}_{3,3} = \begin{array}{c} z^2 f_1 \\ y^z f_1 \\ x^z f_1 \\ y^2 f_1 \\ xy f_1 \\ x^2 f_1 \\ z f_2 \\ y f_2 \\ z f_3 \\ y f_3 \end{array} \begin{array}{c} x^3 \ x^2 y \ xy^2 \ y^3 \ x^2 z \ xyz \ y^2 z \ xz^2 \ yz^2 \ z^3 \\ \left| \begin{array}{cccccccccc} & & & & & & 2 & & 1 & \\ & & & & & 2 & & & 1 & \\ & & & & 0 & & -2 & 1 & & \\ & & & 2 & & & 1 & & & \\ & 0 & & -2 & & 1 & & & 4 & \\ 2 & & & & 1 & & & & & \\ & & & & 1 & & 1 & & & -2 \\ & 1 & & 1 & & & & & -2 & \\ & & & & 0 & & 0 & & 1 & 7 \\ & & & 0 & & & 0 & & 0 & 105 \end{array} \right| \end{array} \begin{array}{l} +O(2^{10}) \\ +O(2^{10}) \\ +O(2^{10}) \\ +O(2^{10}) \\ +O(2^{10}) \\ +O(2^{10}) \\ +O(2^{10}) \\ +O(2^{10}) \\ +O(2^8) \\ +O(2^8) \end{array}.$$

Concernant la précision, nous pouvons par exemple, remarquer sur la dernière matrice que, alors que notre résultat de majoration de la perte de précision sur l'échelonnement d'une matrice (Proposition 1.2.4) nous majore la perte de précision par 6, nous n'observons, du fait du caractère creux de la matrice, qu'une perte de précision de 2.

7.2.3. Précision vs complexité

Afin d'atteindre une plus faible perte de précision lors de l'échelonnement des matrices par pivot de Gauss, nous proposons l'algorithme Matriciel-Faible suivant :

- Calculer les $\mathcal{M}_{d,i}$ comme avant, avec le critère F5 ;
- Au lieu de calculer la forme échelonnée de $\mathcal{M}_{d,i}$, nous calculons la forme échelonnée de la matrice complète $Mac_d(f_1, \dots, f_i)$, jusqu'à la première colonne sans pivot non-nul ;
- Ensuite, on construit la matrice $\widetilde{\mathcal{M}}_{d,i}$ en remplissant la matrice $\mathcal{M}_{d,i}$ en cours de construction avec des multiples des lignes de $Mac_{d-1,i}$, de manière à obtenir une matrice sous forme échelonnée.

Le nombre suivant définit une précision suffisante pour calculer des D -bases de Gröbner avec cet algorithme.

Définition 7.2.16. Soit

$$\square_{d,i} = \min \left(val \left(\left\{ \begin{array}{l} \text{mineur sur les } l_{d,i}\text{-premières} \\ \text{colonnes } Mac_d(f_1, \dots, f_i) \end{array} \right\} \right) \right).$$

Nous définissons la précision de Macaulay de (f_1, \dots, f_s) par rapport à ω et D comme :

$$prec_{Mac}(\{f_1, \dots, f_s\}, D, \omega) = \max_{d \leq D, 1 \leq i \leq s} \square_{d,i}.$$

En effet, avec la Proposition 1.2.7, $prec_{Mac}((f_1, \dots, f_s), D, \omega)$ est suffisant pour calculer des D -bases de Gröbner approchées de suites de polynômes homogènes satisfaisant **H1** et **H2**, et l'Algorithme Matriciel-Faible atteint la meilleure perte de précision que l'échelonnement par pivot de Gauss sur les matrices de Macaulay peut atteindre. Nous remarquons que $prec_{Mac} \leq prec_{MF5}$.

Nous pouvons illustrer sur un exemple comment $prec_{Mac}$ peut être strictement plus petit que $prec_{MF5}$: soit $f = (5x, y, 25xy + z^2)$ dans $\mathbb{Q}_5[x, y, z]$. Alors $prec_{MF5}(f, 2, grevlex(x > y > z)) = 3$ tandis que $prec_{Mac}(f, 2, grevlex(x > y > z)) = 2$.

Cependant, cet algorithme a un prix plus élevé concernant la complexité : échelonner une matrice de Macaulay $Mac_d(f_1, \dots, f_i)$ jusqu'à la première colonne sans pivot est en $O\left(\binom{n+d-1}{d}^2 \times i \binom{n+d-1}{d}\right)$.

Ceci amène à une complexité totale en $O\left(s^2 D \binom{n+D-1}{D}^3\right)$ opérations sur R à précision m , pour $D \rightarrow +\infty$, tandis qu'en utilisant le critère F5 pour construire les matrices, nous avons seulement besoin de $O\left(s D \binom{n+D-1}{D}^3\right)$. Pour résumer :

Théorème 7.2.17. Soit $(f_1, \dots, f_s) \in A^s$ des polynômes homogènes satisfaisant **H1** et **H2**. Soit (f'_1, \dots, f'_s) des approximations des f_i avec précision m sur leurs coefficients. Alors, si m est assez grand, une D -base de Gröbner approchée de (f'_1, \dots, f'_s) par rapport à ω est bien définie. Elle peut être calculée par l'algorithme Matriciel-Faible.

Soit $\text{prec}_{\text{Mac}} = \text{prec}_{\text{Mac}}(\{f_1, \dots, f_s\}, D, \omega)$. Alors, si les f_i sont dans B , les connaître à la précision $m \geq \text{prec}_{\text{Mac}}$ est suffisant, et la perte de précision est majorée par prec_{Mac} . La complexité est en $O\left(s^2 D \binom{n+D-1}{D}^3\right)$ opérations dans R à précision m , pour $D \rightarrow +\infty$.

7.3. Topologie et optimalité

7.3.1. Continuité et optimalité

Nous pouvons réinterpréter le Théorème 7.1.1 de la façon suivante : l'application $\Phi : A_{d_1} \times \dots \times A_{d_s} \rightarrow \mathcal{P}(A)$ qui envoie $f = (f_1, \dots, f_s)$ sur l'ensemble $LM(\langle f_1, \dots, f_s \rangle)$ (l'idéal des termes de tête) est localement constante, et même plus précisément, est constante sur un voisinage de chaque suite satisfaisant **H1** et **H2**. Ce sont ces propriétés qui permettent d'avoir une stabilité numérique au voisinage de f . On pourrait montrer qu'un résultat similaire serait aussi vrai pour $K = \mathbb{R}$, mais dans ce cas, il serait beaucoup plus difficile de trouver un voisinage explicite de f puisque l'on ne pourrait plus appliquer la Proposition 1.1.2 et le Théorème 1.2.6.

Concernant l'optimalité de ces hypothèses de structure, nous remarquons que sans les hypothèses **H1** ou **H2**, le fait que les termes de tête de l'idéal des termes de tête soit localement constant n'est plus nécessairement satisfait. Par exemple, dans $K[X, Y, Z]$, $F = (X + Y, XY + Y^2 + Z^2)$ satisfait **H1** et pas **H2** pour l'ordre lexicographique, et nous pouvons considérer les approximations $F_n = (X + (1 + \pi^n)Y, XY + (1 - \pi^n)Y^2 + Z^2)$, qui intersectent n'importe quel voisinage de F . Pour l'ordre lexicographique (avec $X > Y > Z$), nous avons $LM(\langle F \rangle) = \langle X, Z^2 \rangle$, et ainsi, F ne satisfait pas **H2**. Pour les approximations F_n , nous avons $LM(\langle F_n \rangle) = \langle X, Y^2 \rangle$. Ainsi, pour tout n , $LM(\langle F_n \rangle) \neq LM(\langle F \rangle)$. De la même manière, $f = (X + Y, X^2 + XY)$ satisfait **H2** et pas **H1**, avec le même problème.

7.3.2. Différentiabilité

Calcul de la différentielle

Pour essayer d'appliquer les idées de la précision différentielle, nous allons d'abord calculer la différentielle du calcul de bases de Gröbner réduites.

Soit $f = (f_1, \dots, f_s)$ un élément de $A_{d_1} \times \dots \times A_{d_s}$ satisfaisant **H1** et **H2**. Soit U un voisinage ouvert de f dans $A_{d_1} \times \dots \times A_{d_s}$, donné par exemple par $\text{prec}_{\text{Mac}}(f)$. Soit r le cardinal de la base de Gröbner réduite de $\langle f \rangle$, et d le maximum des degrés d'un élément de cette base de Gröbner réduite. Nous avons alors le résultat suivant :

Proposition 7.3.1. L'application Ψ qui envoie un élément a de U sur la base de Gröbner réduite de $\langle a \rangle$ est une application différentiable¹ de U dans $A_{\leq d}^r$.

Nous pouvons maintenant essayer d'appliquer le Lemme 2.2.4. Pour cela, calculons la différentielle correspondante :

Théorème 7.3.2. Soit $g \in A_{\leq d}^r$ la base de Gröbner réduite de $\langle f \rangle$, et $M \in A^{s \times r}$ tel que $g = f \times M$. Alors, on peut exprimer la différentielle de Ψ en f de la manière suivante : pour tout $\delta f \in A_{d_1} \times \dots \times A_{d_s}$,

$$\Psi'(f) \cdot \delta f = \delta f \times M \mod g.$$

Démonstration. Nous développons au premier ordre $g + \delta g = (f + \delta f) \times (M + \delta M)$. On obtient $\delta g = \delta f \times M + f \times \delta M$. Si δf est assez petit, e.g. δf est tel que $f + \delta f \in U$, alors g et $g + \delta g$ ont les mêmes termes de tête. Comme g est une base de Gröbner réduite, cela implique qu'aucun des $LM(g_i)$ ne divise un terme de δg . En conséquence,

$$\delta g = \delta f \times M \mod g.$$

1. en coordonnées, elle est même rationnelle.

7. Algorithme F5-Matriciel et stabilité

En outre, $f \times \delta M \in \langle f \rangle$. Ainsi, $f \times \delta M = 0 \pmod{g}$. En conclusion, $\delta g = \delta f \times M \pmod{g}$. \square

Remarque 7.3.3. Même si le calcul de $\delta f \times M \pmod{g}$ fournit une manière assez pratique d'étudier le calcul de bases de Gröbner au voisinage de f , l'hypothèse de surjectivité du Lemme 2.2.4 semble difficile à déterminer. Elle reste raisonnable si l'on se ramène à la dimension 1 en se projetant sur chaque coordonnée.

Illustration

Nous fournissons ici un exemple explicite pour comprendre le Théorème 7.3.2 et ses implications qualitatives. Soit $f = (x, xy^2 + y^3 + z^3)$ dans $\mathbb{Q}_p[x, y, z]$. Alors une base de Gröbner réduite de f pour l'ordre grevlex (avec $x > y > z$) est : $g = (x, y^3 + z^3)$ avec $M = \begin{bmatrix} 1 & -y^2 \\ 0 & 1 \end{bmatrix}$.

Maintenant, soit $\delta f = (O(p^5)x, O(p^5)xy^2 + O(p^5)y^3 + O(p^5)z^3)$. Alors une base de Gröbner approchée de $f + \delta f$ est donnée par $(x, y^3 + (1 + O(p^5))z^3)$. Dans ce cas, $\delta g = (0, O(p^5)z^3)$.

En même temps, $\delta f \times M = (O(p^5)x, O(p^5)xy^2 + O(p^5)y^3 + O(p^5)z^3)$. Donc, $\delta f \times M \pmod{g} = (0, O(p^5)z^3)$, et nous avons bien $\delta f \times M \pmod{g} = \delta g$, même si l'hypothèse de surjectivité n'est pas remplie.

7.4. Implémentation

7.4.1. Calculs directs

Une implémentation jouet en [S⁺11] des algorithmes précédents est disponible à l'adresse suivante : http://perso.univ-rennes1.fr/tristan.vaccon/toy_F5.py.

Le but de cette implémentation est le l'étude de la précision. Elle n'est en particulier pas optimisée du point de vue du temps de calcul.

Nous avons expérimenté l'Algorithme F5-Matriciel-Faible jusqu'au degré D (donné par la borne de Macaulay) sur des polynômes homogènes f_1, \dots, f_s , de degrés d_1, \dots, d_s dans $\mathbb{Z}_p[X_1, \dots, X_n]$, avec coefficients pris aléatoirement dans \mathbb{Z}_p avec précision initiale 30. Cette expérience est répétée n_{exp} fois, avec pour ordre monomial grevlex. **max** correspond à la perte de précision maximale observée sur les coefficients des bases obtenues en sortie, en-dehors des cas où la précision était insuffisante. \overline{m} est la partie de précision moyenne sur ces coefficients, **gap** est le maximum des différences pour une expérience entre la perte maximale vue en pratique et la borne théorique $prec_{F5M}$. **f** est le nombre d'échecs (précision insuffisante ou polynômes en entrée ne vérifiant pas **H1** ou **H2**). Nous présentons les résultats dans le tableau suivant :

| d | D | p | n_{exp} | max | \overline{m} | gap | f |
|-----------|----|-----|-----------|------------|----------------|------------|----------|
| [3,4,7] | 12 | 2 | 30 | 11 | .5 | 141 | 0 |
| [3,4,7] | 12 | 7 | 30 | 2 | 0 | 42 | 0 |
| [2,3,4,5] | 11 | 2 | 20 | 25 | 2.2 | 349 | 3 |
| [2,3,4,5] | 11 | 7 | 20 | 5 | .3 | 84 | 0 |
| [2,4,5,6] | 14 | 2 | 20 | 28 | 3.1 | 581 | 3 |
| [2,4,5,6] | 14 | 7 | 20 | 14 | .4 | 73 | 0 |

Ces résultats suggèrent que la perte de précision est moins importante lorsqu'on travaille avec des nombres premiers p plus grand. Cela semble raisonnable puisque les pertes de précision proviennent de pivots avec des valuation strictement positives, tandis que, pour la mesure de Haar sur \mathbb{Z}_p , la probabilité que $val(x) = 0$ pour $x \in \mathbb{Z}_p$ est $\frac{p-1}{p}$. De même la perte de précision semble augmenter avec la taille des matrices de Macaulay considérées dans l'algorithme.

Concernant l'écart entre $prec_{F5M}$ et la perte de précision constatée en pratique, nous remarquons que la majoration donnée par $prec_{F5M}$ est obtenue par le Théorème 1.2.6 qui est montré pour des matrices denses. Une grande valuation pour un pivot se répercute dans $prec_{MF5}$, même si elle ne générerait pas de perte de précision en pratique si n'il y a pas de coefficient non-nul à éliminer sur la colonne du pivot. Ainsi, le Théorème 1.2.6 ne tient pas compte du caractère creux des matrices de Macaulay, ce qui explique qualitativement pourquoi **gap** est si grand comparé à **max**.

7.4.2. De la stabilité

Grâce au Théorème 7.3.2, nous avons maintenant à notre disposition trois manières d'étudier la perte de précision lors du calcul de bases de Gröbner réduites :

- **Calcul direct** : dans $\mathbb{Z}_p[X_1, \dots, X_n]$, à travers l'Algorithme 7.2.7, en partant de polynômes dont les coefficients sont connus avec précision $O(p^k)$.
- **Méthode des Différences** : Calculer les bases de Gröbner réduites $g^{(1)}$ de $f^{(1)} \in \mathbb{Z}[X_1, \dots, X_n]^s$ et $g^{(2)}$ de $f^{(2)} = f^{(1)} + df$, pour des $df \in (p^k \mathbb{Z}[X_1, \dots, X_n])^s$, regarder les valuations p -adiques des coefficients de $g^{(1)} - g^{(2)}$. Cette méthode ne permet cependant que de se donner une idée du résultat ;
- **Différentielle** : Calculer la différentielle du calcul de la base de Gröbner réduite en $f^{(1)}$ avec le Théorème 7.3.2.

Nous avons comparé ces trois méthodes dans le même contexte que pour la Sous-Section précédente (polynômes homogènes de degrés donnés avec coefficients pris aléatoirement et précision initiale 30). La liste dans la colonne Min montre, pour chacune des n_{exp} expériences le minimum de la précision soit obtenue sur la base de Gröbner réduite dans le cas directe, soit donnée par l'estimation de la perte de précision théorique par les méthodes de différences et différentielles.

| d | D | p | n_{exp} | méthode | Min |
|---------|---|-----|-----------|----------------|--|
| [2,2,3] | 5 | 2 | 10 | directe | [28, 24, 6, 15, 25, 27, 22, 22, 26, 22] |
| [2,2,3] | 5 | 2 | 10 | différence | [30, 25, 6, 18, 25, 30, 28, 24, 27, 22] |
| [2,2,3] | 5 | 2 | 10 | différentielle | [30, 25, 6, 18, 25, 30, 28, 24, 27, 22] |
| [2,2,3] | 5 | 7 | 10 | directe | [29, 26, 28, 29, 29, 30, 26, 30, 28, 30] |
| [2,2,3] | 5 | 7 | 10 | différence | [30, 26, 29, 30, 29, 30, 27, 30, 28, 30] |
| [2,2,3] | 5 | 7 | 10 | différentielle | [30, 26, 29, 30, 29, 30, 26, 30, 28, 30] |
| [2,3,4] | 7 | 2 | 10 | directe | [22, 26, 28, 27, 24, 23, 27, 18, 21, 22] |
| [2,3,4] | 7 | 2 | 10 | différence | [23, 26, 29, 28, 26, 29, 28, 20, 26, 22] |
| [2,3,4] | 7 | 2 | 10 | différentielle | [23, 26, 29, 28, 26, 29, 28, 20, 26, 22] |
| [2,3,4] | 7 | 7 | 10 | directe | [30, 28, 28, 30, 28, 30, 28, 28, 26, 30] |
| [2,3,4] | 7 | 7 | 10 | différence | [30, 29, 29, 30, 28, 30, 28, 28, 26, 30] |
| [2,3,4] | 7 | 7 | 10 | différentielle | [30, 28, 28, 30, 28, 30, 28, 28, 26, 30] |

De ce tableau, nous pouvons suggérer les heuristiques suivantes :

- Même si l'hypothèse de surjectivité n'est pas garantie, la différentielle donne une estimation de la perte de précision très proche de ce que donne la méthode des différences, et ainsi de la perte de précision intrinsèque ;
- Nos calculs directs se révèlent souvent stables, avec numériquement une perte de précision proche de ce qui est attendu, mais, comme on pouvait s'y attendre, arrivent parfois à des pertes de précision plus grandes que la perte de précision intrinsèque donnée théoriquement.

7.5. Méthode de remontée sous l'hypothèse **H2**

7.5.1. Remonter une base de Gröbner

Dans cette Sous-Section, nous considérons le problème suivant : dans la Définition 7.2.1 d'une base de Gröbner approchée, nous énonçons que pour toute spécialisation des $O(\pi^n)$ des coefficients des polynômes en entrée, il y a une spécialisation des $O(\pi^n)$ des coefficients des polynômes de la base de Gröbner approchée qui est cohérente avec la première spécialisation. En d'autres mots, c'est une base de Gröbner de l'idéal engendré par les polynômes en entrée. Cependant, si l'on connaît une base de Gröbner approchée à une certaine précision et que l'on donne de nouveaux chiffres de précision aux polynômes en entrée (une spécialisation), y a-t-il une manière plus intelligente de voir comment ces nouveaux chiffres de précision se répercutent sur la base de Gröbner approchée sans avoir à refaire le calcul complet d'une base de Gröbner approchée ?

Une première idée naturelle serait d'adapter une méthode de Newton-Hensel. Utiliser une adaptation du lemme de Hensel pour accélérer le calcul des bases de Gröbner a été proposé pour la première fois dans [Win88], et cette idée a été poursuivie ou utilisée dans [Pau92], [Arn03] et [RY06]. L'idée principale est qu'étant donné $f = (f_1, \dots, f_s) \in \mathbb{Z}[X_1, \dots, X_n]$, on calcule d'abord la base de

Gröbner réduite $\bar{g} = (\bar{g}_1, \dots, \bar{g}_r)$ dans $\mathbb{Z}/p\mathbb{Z}[X_1, \dots, X_n]$ de la réduction modulo p de f , et ensuite une méthode de Newton-Hensel pour obtenir la base de Gröbner réduite de $g = (g_1, \dots, g_r)$ dans $\mathbb{Z}[X_1, \dots, X_n]$ de $\langle f \rangle$. Pour que cette méthode fonctionne, on requiert souvent une hypothèse de "bonne fortune"² sur p , *e.g.* que \bar{g} est la réduction modulo p de g . Ces conditions sont en général très difficiles à vérifier *a priori*. Nous montrons dans cette Sous-Section que sous les hypothèses **H1** et **H2**, on peut effectuer directement une méthode de remontée sur une base de Gröbner approchée (étant donnée l'écriture de ses polynômes en fonction des polynômes en entrée). L'hypothèse de "bonne fortune" sur p est alors remplacée par une hypothèse de précision suffisante.

7.5.2. Remontée à des points satisfaisant H1 et H2

Présentation de l'algorithme

L'objectif principal de l'Algorithme 7.5.2 est le suivant. Nous disposons de la famille de polynômes homogènes en entrée $F \in B^s$ et d'une D -base de Gröbner approchée $G \in B^r$ avec une matrice, connue elle aussi de manière approchée, $M \in B^{s \times r}$ telle que $G = (F + O(\pi^m)) \cdot M$. Nous voulons alors calculer une D -base de Gröbner approchée pour $F + O(\pi^l)$ avec $l > m$. Si F peut être connu à précision infinie (*e.g.* $F \in \mathbb{Q}[X_1, \dots, X_n]^s$), nous pouvons obtenir par le même algorithme une D -base de Gröbner de F . Ceci correspond à $l = +\infty$ dans ce qui suit.

Pour cela, nous allons effectuer ce que nous appelons une remontée canonique de M . Par ce terme, nous voulons dire que nous ajoutons des chiffres 0 dans le développement π -adique des coefficients de M jusqu'à la précision $O(\pi^l)$.

L'idée principale est alors la suivante. Nous effectuons une remontée canonique de M pour obtenir une matrice \widehat{M} dont les coefficients sont connus à précision $O(\pi^l)$, et nous calculons $H = (F + O(\pi^l)) \times \widehat{M}$. La D -base de Gröbner approchée recherchée s'obtient alors en effectuant une inter-réduction sur H .

Remarque 7.5.1. Dans l'Algorithme 7.5.2, nous utilisons l'Algorithme 7.2.7 avec la simple modification que celui-ci calcule une matrice M telle que $G = F \times M$ en même temps que le calcul de G .

Algorithme 7.5.2 : L'algorithme de Remontée-Faible

Entrée : $F = (f_1, \dots, f_s) \in R[X_1, \dots, X_n]^s$, polynômes homogènes de degrés respectifs $d_1 \leq \dots \leq d_s$, et $D \in \mathbb{N}$, un ordre monomial ω .
La précision m utilisée dans le premier calcul et la précision l à laquelle effectuer la remontée.
Sortie : $(g_1, \dots, g_k) \in A^k$, une D -base de Gröbner approchée de $\langle F \rangle$, ou **Erreur** si (f_1, \dots, f_s) ne satisfait pas **H1**, **H2** ou que la précision n'est pas suffisante.

début

```

     $G, M \leftarrow \text{weak-MF5}(F + O(\pi^m));$  // Nous avons  $G = (F + O(\pi^m)) \cdot M$ 

     $\widehat{M} \leftarrow$  remontée canonique de  $M$  jusqu'à la précision  $O(\pi^l)$ ;
     $H \leftarrow (F + O(\pi^l)) \cdot \widehat{M}$ ;
     $\widehat{G} \leftarrow []$ ;
    pour  $i \in [1, \#H]$  faire
         $\widehat{G}.\text{Ajouter}(H[i] \bmod \widehat{G})$ ;
        /* Nous supposons que les  $H[i]$  sont ordonnés degré par degré, et triés par ordre
           décroissant à degré fixé selon leurs termes de têtes */
    Retourner  $\widehat{G}$ .
```

Correction

Nous prouvons ici que l'Algorithme 7.5.2 calcule bien une D -base de Gröbner approchée par remontée.

2. L'auteur s'excuse de n'avoir pas de meilleure traduction à proposer pour "luckiness"

Soit $f = (f_1, \dots, f_s) \in B_{d_1} \times \dots \times B_{d_s}$ satisfaisant **H1** et **H2**. Soit n un entier définissant une précision suffisante sur f . Soit $G = (g_1, \dots, g_r) \in B^r$ une D -base de Gröbner approchée de $\langle f + O(\pi^m) \rangle$, avec $M \in B^{s \times r}$ polynômes homogènes, tels que pour un certain $l_1 > 0$, nous avons $g + O(\pi^{l_1}) = (f + O(\pi^m)) \times (M + O(\pi^m))$. Nous supposons que les g_i sont triés de telle manière que $|g_i| < |g_{i+1}|$, ou $|g_i| = |g_{i+1}|$ et $LM(g_i) > LM(g_{i+1})$. La sortie de l'Algorithme 7.2.7, F5-Matriciel-Faible satisfait cette hypothèse.

Soit $l > m$, ou $l = +\infty$ dans le cas où f peut être donné à précision infinie. Soit \widehat{M} la remontée canonique de $M + O(\pi^m)$ jusqu'à la précision $O(\pi^l)$, i.e. $\widehat{M} = M + O(\pi^m)$ et les chiffres dans le développement π -adique des coefficients de \widehat{M} qui sont compris entre π^m et π^l sont des zéros. Alors le Lemme suivant montre la correction de l'Algorithme 7.5.2 :

Lemme 7.5.3. *Soit $H = (f + O(\pi^m)) \times (\widehat{M} + O(\pi^m))$. Alors, pour m et l_1 assez grands, \widehat{G} donné par la réduction successive des h_i par les $(\widehat{g}_1, \dots, \widehat{g}_{i-1})$ est une D -base de Gröbner approchée de $\langle f + O(\pi^m) \rangle$. Si G est la sortie de l'Algorithme 7.2.7, alors $n > 2\text{prec}_{MF5}(f)$ définit une précision suffisante et $l_1 = n - \text{prec}_{MF5}(f)$. Si G est une base de Gröbner réduite à renormalisation des coefficients de têtes près, alors il en est de même pour \widehat{G} .*

Démonstration. Nous remarquons tout d'abord que $H = G + O(\pi^{l_1})$, et H est constituée de polynômes homogènes car il en est de même pour \widehat{M} et F . Nous prouvons ensuite par récurrence que pour tout i , $LM(\widehat{g}_i) = LM(g_i)$.

À cause de l'hypothèse **H2**, $LM(g_1)$ est le plus grand monôme de degré $|g_1|$. Puisque $h_1 = \widehat{g}_1$ est homogène du même degré que g_1 et $\widehat{g}_1 = g_1 + O(\pi^{l_1})$, $LM(\widehat{g}_1) = LM(g_1)$. Supposons maintenant que pour un certain $i > 0$, nous avons pour tout j tel que $1 \leq j < i$, $LM(\widehat{g}_j) = LM(g_j)$. Soit $x^\beta = LM(g_i)$. Alors, l'hypothèse **H2** signifie que pour tout $x^\alpha > x^\beta$ et de degré $|g_i|$, il y a un $LM(\widehat{g}_j)$ avec $j < i$ et qui le divise. Ainsi, $LM(\widehat{g}_i) \leq LM(g_i)$. Néanmoins, les coefficients en x^α de h_i , pour $x^\alpha > x^\beta$ et de degré $|g_i|$, sont dant $\pi^{l_1}R$. En conséquence, il existe un $c \geq 0$ tel que, après réduction de h_i par les \widehat{g}_j , les coefficients en x^β de g_i et \widehat{g}_i sont égaux modulo π^{l_1-c} . Ainsi, si l_1 est assez grand, ceci implique que ce coefficient est non-nul et ainsi que $LM(\widehat{g}_i) = LM(g_i)$.

Dans le cas particulier où G est le résultat de l'Algorithme 7.2.7 et $n > 2\text{prec}_{MF5}(f)$, $l_1 = m - \text{prec}_{MF5}(f)$, alors $c \leq \text{prec}_{MF5}$ et l_1 est effectivement assez grand. Le résultat est donc prouvé.

Pour ce qui est des bases de Gröbner réduites, comme $LM(\widehat{g}_i) = LM(g_i)$ et avec la définition de $LM(\widehat{g}_i)$, le résultat est clair. \square

Heuristiquement, n et l_1 sont assez grands lorsque G peut supporter une *deuxième* réduction (en lignes) pour certifier ses monômes de tête. C'est pourquoi, si l'on applique d'abord l'Algorithme 7.2.7, $n > 2\text{prec}_{F5M}$ est suffisant.

Complexité

Nous pouvons maintenant estimer la complexité de l'Algorithme 7.5.2 :

Proposition 7.5.4. *La complexité de l'Algorithme 7.5.2 est en $O\left(s^2 D^{(n+D-1)^3}\right)$ opérations dans K à précision m et $O\left((s + \#G)^{(n+D-1)^2}\right)$ opérations à précision l , pour $D \rightarrow +\infty$.*

Démonstration. Le calcul de M en même temps que g ajoute un facteur s à la complexité asymptotique du calcul d'une D -base de Gröbner, ce qui correspond à une complexité totale en $O\left(s^2 D^{(n+D-1)^3}\right)$ opérations dans K à la précision initiale m . En effet, il est suffisant d'ajouter aux lignes des matrices de Macaulay des étiquettes exprimant le polynôme de chaque ligne en fonction des polynômes en entrée, et de répercuter les opérations sur les lignes sur ces étiquettes. Ceci ajoute bien un facteur s à la complexité du calcul d'une opération sur les lignes et ainsi, donne la complexité totale pour le calcul de M et g . La complexité du calcul de H est alors en $O\left(s^{(n+D-1)^2}\right)$ opérations à précision l , et les inter-réductions pour calculer \widehat{G} sont en $O\left(\#G^{(n+D-1)^2}\right)$ opérations à précision l . \square

En conséquence, dans l'Algorithme 7.5.2, le coût de l'algèbre linéaire est complètement porté par l'arithmétique à la précision finie initiale. En particulier, si l est très grand devant m , la

seconde étape peut être la plus coûteuse, mais le nombre d'opérations à précision l n'est plus qu'en $O\left(\binom{n+D-1}{D}^2\right)$ par rapport à D et n , soit moins que lors d'un calcul direct.

7.5.3. Application

Nous remarquons que dans le Lemme 7.5.3, si les entrées peuvent être connues à précision infinie, alors on peut faire une remontée sur la base de Gröbner approchée jusqu'à une précision infinie. Ceci implique que si les polynômes en entrée $f = (f_1, \dots, f_s)$ ont leurs coefficients dans \mathbb{Q} , et que f satisfait **H1** et **H2**, il est alors possible d'appliquer l'Algorithme 7.5.2 en calculant une première base de Gröbner, approchée, de $\langle f \rangle$ à une précision juste suffisamment grande, et ensuite en la remontant une base de Gröbner (à précision infinie). Grâce au fait que l'essentiel de l'algèbre linéaire est effectuée à précision finie, ceci amène à une complexité totale qui peut être vue comme intermédiaire entre le calcul de bases de Gröbner pour des polynômes ayant leurs coefficients dans un corps finis et celle de calcul de bases de Gröbner pour des polynômes ayant leurs coefficients dans \mathbb{Q} .

Ce résultat peut aussi être vu comme une réponse au problème du calcul de nombres premiers "fortunés"³. Si l'on prend p de taille moyenne, par exemple 7 et que l'on travaille à une précision faible, comme 11, alors cette précision était suffisante pour appliquer l'Algorithme 7.5.2 dans les 20 cas testés pour $d = [2, 3, 4, 5]$ dans la Sous-Section 7.4. Une précision 30 était suffisant dans les 20 cas testés pour $d = [2, 4, 5, 6]$, et dans la plupart des cas, seule une précision plus faible était nécessaire. Nous pouvons aussi prendre un p plus grand, ce qui permet de travailler à précision plus faible.

En outre, du fait de la continuité dans la Proposition 7.3.1, quitte à ne pas pouvoir certifier le résultat dans un premier temps, ni avoir une bonne estimation de la perte de précision, l'appel à l'Algorithme F5-Matriciel-Faible dans l'Algorithme 7.5.2 peut être remplacé par n'importe quel algorithme de calcul de bases de Gröbner (*e.g.* les algorithmes F4 ou F5 de Faugère), pour obtenir une meilleure complexité dans le calcul de la première base de Gröbner approchée.

Enfin, nous illustrons cette stratégie sur un exemple explicite. Soit $f = (10x, 25xy^2 + y^3 + z^3)$ dans $\mathbb{Q}[x, y, z]$. Nous travaillons d'abord avec $\tilde{f} = (10 + O(5^4))x, (25 + O(5^4))xy^2 + (1 + O(5^4))y^3 + (1 + O(5^4))z^3$ in $\mathbb{Q}_5[x, y, z]$ à précision initiale 4. Alors l'Algorithme 7.2.7 fournit, après élimination des lignes sans monôme de tête minimal, la base de Gröbner approchée suivante $G = ((10 + O(5^4))x, (1 + O(5^3))y^3 + (1 + O(5^3))z^3)$, avec $M = \begin{bmatrix} 1 & -(3 \cdot 5 + 2 \cdot 5^2 + O(5^3))y^2 \\ 0 & 1 \end{bmatrix}$.

Ceci amène $\widehat{M} = \begin{bmatrix} 1 & -65y^2 \\ 0 & 1 \end{bmatrix}$, qui, avec f , donne $H = (10x, -5^4xy^2 + y^3 + z^3)$. L'Inter-réduction donne finalement la base de Gröbner minimale (et réduite à renormalisation des coefficients de tête près) $\widehat{G} = (10x, y^3 + z^3)$.

7.6. Le cas affine et base de Gröbner réduite

7.6.1. Le cas affine

Une des principales restrictions de l'Algorithme F5-Matriciel-Faible 7.2.7 est qu'il demande à ce qu'en entrée, les polynômes soient homogènes. Or, cette restriction peut être, génériquement, évitée. En effet, lorsque l'ordre monomial considéré raffine l'ordre du degré, il est possible d'étendre le Théorème 7.1.1 et l'Algorithme 7.2.7 à des polynômes en entrée non-homogènes par une méthode similaire à celle que l'on peut trouver dans [FSEDV13] or [FGHR14]. Il suffit de faire porter l'hypothèse **H1** sur les composantes homogènes de plus haut degré, et de faire le calcul d'une base de Gröbner de l'idéal engendré par ces composantes homogènes de plus haut degré, et on pourra ensuite récupérer une base de Gröbner de l'idéal.

Le théorème suivant est en effet bien connu :

Théorème 7.6.1. *Soit ω un ordre monomial sur $K[X_1, \dots, X_n]$ qui raffine l'ordre du degré. Soit $(f_1, \dots, f_s) \in K[X_1, \dots, X_n]^s$ des polynômes. Soit f_1^h, \dots, f_s^h leurs composantes homogènes de plus*

3. "lucky" primes

haut degré. Nous supposons que (f_1^h, \dots, f_s^h) satisfait **H1** (i.e. est une suite régulière). Alors

$$LM(\langle f_1, \dots, f_s \rangle) = LM(\langle f_1^h, \dots, f_s^h \rangle)$$

Démonstration. Nous renvoyons à la Proposition 13 de [FGHR14]. \square

Ce théorème est alors suffisant pour étendre notre étude au cas affine :

Proposition 7.6.2. *Soit ω un ordre monomial sur $K[X_1, \dots, X_n]$ qui raffine l'ordre du degré. Soit $(f_1, \dots, f_s) \in K[X_1, \dots, X_n]^s$ des polynômes. Soit f_1^h, \dots, f_s^h leurs composantes homogènes de plus haut degré. Nous supposons que (f_1^h, \dots, f_s^h) satisfait **H1** et **H2**. Soit (f'_1, \dots, f'_s) des approximations des f_i avec précision m sur leurs coefficients. Alors, si m est assez grand, une D -base de Gröbner approchée de (f'_1, \dots, f'_s) par rapport à ω est bien définie.*

De plus, si les f_i sont dans $R[X_1, \dots, X_n]$, alors $m \geq \text{prec}_{F5M}(f_1^h, \dots, f_s^h)$ est suffisant, et la perte de précision est majorée par $\text{prec}_{F5M}(f_1^h, \dots, f_s^h)$ ou $\text{prec}_{Mac}(f_1^h, \dots, f_s^h)$.

Démonstration. Grâce au Théorème 7.6.1, $LM(\langle f_1^h, \dots, f_s^h \rangle) = LM(\langle f_1, \dots, f_s \rangle)$. Soit $(h_1, \dots, h_r) \in K[X_1, \dots, X_n]^r$ une base de Gröbner de $\langle f_1^h, \dots, f_s^h \rangle$, constituée de polynômes homogènes $h_i = \sum_j a_{i,j} f_j^h$, pour quelques $a_{i,j} \in K[X_1, \dots, X_n]$, homogènes, et $r \geq 0$. Soit $(g_1, \dots, g_r) \in K[X_1, \dots, X_n]^r$ tel que $g_i = \sum_j a_{i,j} f_j$. Alors en conséquence, (g_1, \dots, g_r) est une base de Gröbner de $\langle f_1, \dots, f_s \rangle$.

Ceci implique que le calcul d'une base de Gröbner de $\langle f_1, \dots, f_s \rangle$ peut être totalement déterminé par celui de $\langle f_1^h, \dots, f_s^h \rangle$. Il est alors clair qu'il suffit d'appliquer le Théorème 7.1.1 ou le Théorème 7.2.17 à (f_1^h, \dots, f_s^h) . \square

Définition 7.6.3. Nous disons que $F = (f_1, \dots, f_s)$ une famille de polynômes de $K[X_1, \dots, X_n]$ satisfait **H1a** si $F^h = (f_1^h, \dots, f_s^h)$, la famille de ses composantes homogènes de plus haut degré satisfait **H1**. Nous définissons de même **H2a** avec **H2** pour F^h .

Remarque 7.6.4. Nous pouvons alors énoncer une conjecture de Moreno-Socias affine : si k est un corps infini, $s \in \mathbb{N}$, $d_1, \dots, d_s \in \mathbb{N}$, alors il existe un ouvert de Zariski non-vide U dans $A_{\leq d_1} \times \dots \times A_{\leq d_s}$ tel que pour tout $(f_1, \dots, f_s) \in U$, $I = (f_1, \dots, f_s)$ est un idéal faiblement grevlex. Les conjectures affines et non-affines sont clairement équivalentes. En conséquence, la conjecture de Moreno-Socias implique que les suites $(f_1, \dots, f_s) \in A_{\leq d_1} \times \dots \times A_{\leq d_s}$ satisfaisant **H1a** et **H2a** sont génériques.

7.6.2. Calcul de la base de Gröbner réduite

Nous étudions maintenant la perte de précision lors du calcul d'une base de Gröbner réduite, approchée, à partir d'une base de Gröbner approchée. Pour cela, nous définissons ce que nous appellerons le nombre de condition d'une base de Gröbner, et qui code la perte de précision lors de la division par une base de Gröbner.

Définition 7.6.5. Nous notons $\text{cond}(G, \omega)$ le nombre de condition d'une base de Gröbner G , dont les éléments sont dans B , pour un ordre monomial ω , et le définissons par la formule suivante :

$$\max_{d \leq D} \sum_{x^\alpha \in LM(I) \cap A_d} \text{minvalLC}(x^\alpha),$$

avec D le maximum des degrés des éléments de la base de Gröbner minimale déduite de G , et $\text{minvalLC}(x^\alpha) = \min\{\text{val}(\text{LC}(x^\beta g)) \mid g \in G \text{ et } LM(gx^\beta) = x^\alpha\}$. Si $G = (g_1, \dots, g_t)$ est tel que les g_i sont dans A , nous notons $\text{cond}(G, \omega)$ pour $\text{cond}(G', \omega)$ avec $G' = (c_1 g_1, \dots, c_t g_t)$ où les c_i sont dans R , de plus petite valuation tel que pour tout i $c_i g_i \in B$.

Nous avons alors le résultat suivant :

Proposition 7.6.6. *Soit G une base de Gröbner approchée dans A pour un ordre monomial ω . Alors la perte de précision pour calculer la base de Gröbner réduite de $\langle G \rangle$ pour ω est majorée par $\text{cond}(G, \omega)$.*

Démonstration. Il s'agit simplement d'appliquer la Proposition 1.1.2 lors du calcul de la division d'un élément g de G par $G \setminus \{g\}$. \square

7. Algorithme F5-Matriciel et stabilité

En conséquence, nous pouvons énoncer les résultats suivants concernant la perte de précision lors du calcul d'une base de Gröbner réduite dans le cas homogène :

Théorème 7.6.7. *Soit ω un ordre monomial sur $K[X_1, \dots, X_n]$. Soit $(f_1, \dots, f_s) \in K[X_1, \dots, X_n]^s$ des polynômes. Nous supposons que (f_1, \dots, f_s) satisfait **H1** et **H2**. Soit (f'_1, \dots, f'_s) des approximations des f_i avec précision m sur leurs coefficients. Alors, si m est assez grand, une D -base de Gröbner approchée réduite de (f'_1, \dots, f'_s) par rapport à ω est bien définie.*

De plus, si les f_i sont dans $R[X_1, \dots, X_n]$, nous pouvons préciser la perte de précision. Soit G la base de Gröbner approchée obtenue par l'Algorithme 7.2.7. Alors $m \geq \text{prec}_{F5M}(f_1, \dots, f_s) + \text{cond}(G, \omega)$ est suffisant pour calculer la base de Gröbner approchée réduite de (f'_1, \dots, f'_s) , et la perte de précision est majorée par $\text{prec}_{F5M}(f_1, \dots, f_s) + \text{cond}(G, \omega)$. En appliquant la Sous-Section 7.2.3, on obtient $\text{prec}_{Mac}(f_1, \dots, f_s) + \text{cond}(G, \omega)$.

Ce résultat se généralise naturellement au cas affine :

Théorème 7.6.8. *Soit ω un ordre monomial sur $K[X_1, \dots, X_n]$ qui raffine l'ordre du degré. Soit $(f_1, \dots, f_s) \in K[X_1, \dots, X_n]^s$ des polynômes. Soit f_1^h, \dots, f_s^h leurs composantes homogènes de plus haut degré. Nous supposons que (f_1^h, \dots, f_s^h) satisfait **H1** et **H2**. Soit (f'_1, \dots, f'_s) des approximations des f_i avec précision m sur leurs coefficients. Alors, si m est assez grand, une D -base de Gröbner approchée réduite de (f'_1, \dots, f'_s) par rapport à ω est bien définie.*

De plus, si les f_i sont dans $R[X_1, \dots, X_n]$ nous pouvons préciser la perte de précision. Soit G la base de Gröbner approchée obtenue par l'Algorithme 7.2.7 sur les (f_1^h, \dots, f_s^h) . Alors $m \geq \text{prec}_{F5M}(f_1^h, \dots, f_s^h) + \text{cond}(G, \omega)$ est suffisant, et la perte de précision est majorée par

$$\text{prec}_{F5M}(f_1^h, \dots, f_s^h) + \text{cond}(G, \omega).$$

En appliquant la Sous-Section 7.2.3, on obtient $\text{prec}_{Mac}(f_1^h, \dots, f_s^h) + \text{cond}(G, \omega)$.

8. Stabilité de FGLM

"All your base are belong to us."

Cats, Zero Wing

Ce chapitre est consacré à l'étude sur K un CDVF à précision finie, de l'algorithme FGLM, sous ses deux principales variantes que nous avons présentées dans la Sous-Section 6.2.2. Il s'agit d'un travail en commun avec Guénaél Renault. Nous montrons, dans la Section 8.1, qu'étant donnée une base de Gröbner approchée d'un idéal I de dimension zéro pour un ordre monomial \leq , l'algorithme FGLM peut être utilisé à précision finie pour calculer une base de Gröbner approchée de I pour un autre ordre monomial, \leq_2 , et la précision nécessaire est donnée explicitement par la matrice de changement de base de A/I entre les bases canoniques pour \leq et \leq_2 . Dans la Section 8.2, nous montrons que les variantes plus rapides lorsqu'on est dans un cas générique et que l'on part de grevlex ou va vers lex sont aussi disponibles, avec même condition sur la précision.

8.1. Stabilité de l'algorithme direct

8.1.1. Un algorithme stabilisé

Nous débutons notre étude de l'algorithme FGLM à précision finie en présentant un algorithme, que nous appellerons Algorithme FGLM stabilisé, et qui est adapté à la précision finie. La principale différence avec l'algorithme FGLM classique 6.2.38 est que nous remplaçons l'échelonnement en lignes par un calcul de formes normales de Smith, approchées et exactes, comme définies en Section 1.3. Ceci permet de conserver la perte de précision plus faible obtenue en passant par cette forme

normale, ainsi qu'une étude plus aisée de la perte de précision.

Algorithme 8.1.1 : Algorithme FGLM stabilisé

entrée : Une base de Gröbner approchée réduite G_1 pour un ordre monomial \leq d'un idéal $I \subset A$ de dimension zéro et de degré δ , $B_{\leq} = (1 = \epsilon_1 \leq \epsilon_2 \leq \dots \leq \epsilon_\delta)$ la base canonique de A/I pour \leq .
Un ordre monomial \leq_2 .
sortie : Une base de Gröbner approchée G_2 de I pour \leq_2 , ou **Erreur** si la précision est insuffisante.

début

Calculer les matrices de multiplications T_1, \dots, T_n pour I et \leq avec l'Algorithme 6.2.34 ;
 $B_2 := \{1\}$; $\mathbf{v} = [{}^t(1, \dots, 0)]$; $G_2 := \emptyset$;
 $L := \{(1, n), (1, n-1), \dots, (1, 1)\}$;
 $Q_1, Q_2, P_1, P_2, \Delta := I_1, I_1, I_\delta, I_\delta, \mathbf{v}$;
tant que $L \neq \emptyset$ **faire**
 $m := L[1]$; supprimer m de L ;
 $j := m[1]$; $i := m[2]$;
 $v := T_i \mathbf{v}[j]$;
 $s := \text{card}(B_2)$;
 $\lambda = {}^t(\lambda_1, \dots, \lambda_\delta) := P_1 v$;
 si on n'a pas de chiffres significatif sur $\lambda_{s+1}, \dots, \lambda_\delta$ (i.e. ce sont des $O(\pi^v)$) **alors**
 Calculer la forme normale de Smith de \mathbf{v} à partir de la forme normale de Smith approchée donnée par Δ, P_1, Q_1 et leurs inverses P_2, Q_2 , avec l'Algorithme 1.3.8 ;
 Trouver W tel que $\mathbf{v}W = v$, grâce à la forme normale de Smith de \mathbf{v} , et en supposant que $v \in \text{Im}(\mathbf{v})$ (Proposition 1.3.15);
 $G_2 := G_2 \cup \{B_2[j]x_i - \sum_{l=1}^s W_l B_2[l]\}$
 sinon
 $B_2 := B_2 \cup \{B_2[j]x_i\}$;
 $\mathbf{v} = \mathbf{v} \cup [v]$;
 $L := \text{TriCroissant}(L \cup [(s+1, l) | 1 \leq l \leq n], \leq_2)$;
 Enlever les répétitions dans L ;
 $\text{Update}(\mathbf{v}, s, P_1, P_2, Q_1, Q_2, \Delta)$;
 Enlever de L tous les multiples de $LM_{\leq_2}(G_2)$;
si $\text{card}(B_2) = \delta$ **alors**
 Retourner G_2 ;
sinon
 Retourner **"Erreur, précision insuffisante"**

Algorithme 8.1.2 : Update, forme normale de Smith approchée itérée

entrée : Un entier s . Une matrice \mathbf{v} de taille $\delta \times s$, P_1, Q_1, Δ des matrices telles que $P_1 \mathbf{v}' Q_1 = \Delta$ réalise une **forme normale de Smith approchée** de \mathbf{v}' avec \mathbf{v}' la sous-matrice de \mathbf{v} correspondant à ses $s-1$ premières colonnes. On dispose de P_2, Q_2 , supposés être les inverses de P_1, Q_1 .
sortie : $P_1, P_2, Q_1, Q_2, \Delta$ sont mis à jour de telle manière que $P_1 \mathbf{v}' Q_1 = \Delta$ réalise une forme normale de Smith approchée de \mathbf{v} , et P_2, Q_2 inverses de P_1, Q_1 .

début

Augmenter trivialement les matrices Q_1, Q_2 en des matrices carrés ayant une ligne et une colonne de plus, inversibles ;
Calculer U_1, V_1 et Δ' réalisant une forme normale de Smith approchée de $P_1 \mathbf{v}' Q_1$, ainsi que U_2, V_2 les inverses de U_1, V_1 par l'Algorithme 1.3.5 ;
 $P_1 := U_1 \times P_1$;
 $Q_1 := Q_1 \times V_1$;
 $P_2 := P_2 \times U_2$;
 $Q_2 := V_2 \times Q_2$;
 $\Delta := \Delta'$;

Remarque 8.1.3. Pour la résolution de système linéaire, nous utilisons le calcul d'une forme normale de Smith à partir d'une forme approchée grâce à l'Algorithme 1.3.8, puis la résolution comme dans la Proposition 1.3.15.

La notion suivante de conditionnement d'un idéal pour un changement d'ordre monomial est la grandeur qui nous permet de contrôler le comportement de la précision au cours de l'Algorithme FGLM stabilisé 8.1.1 :

Définition 8.1.4. Soit $I \subset A$ de dimension zéro. Soit \leq et \leq_2 deux ordre monomiaux sur A . Soit B_{\leq} et B_{\leq_2} les bases canoniques de A/I pour les ordres \leq et \leq_2 . Soit M la matrice dont les colonnes sont les $NF_{\leq}(x^\beta)$ pour $x^\beta \in B_{\leq_2}$. Nous définissons le conditionnement de I pour le changement d'ordre monomial de \leq vers \leq_2 , noté $\text{cond}_{\leq, \leq_2}(I)$ (ou $\text{cond}_{\leq, \leq_2}$ s'il n'y pas d'ambiguïté) comme la plus grande valuation d'un facteur invariant de M . Nous noterons $\text{cond}_{\leq, \leq_2}(I)^+ = \max(\text{cond}_{\leq, \leq_2}(I), 0)$.

Remarque 8.1.5. La définition de $\text{cond}_{\leq, \leq_2}(I)^+$ sera utile si l'on souhaite négliger les gains de précision.

Nous pouvons maintenant énoncer le théorème suivant, concernant le comportement à précision finie de l'Algorithme FGLM stabilisé :

Théorème 8.1.6. *Étant donnés une base de Gröbner approchée réduite G_1 , pour un ordre monomial \leq , d'un idéal $I \subset A$ de dimension zéro et de degré δ , et un ordre monomial \leq_2 , soit $\text{cond}_{\leq, \leq_2}$ le conditionnement de I pour le changement d'ordre monomial de \leq vers \leq_2 . Alors, si les coefficients des polynômes de G_1 sont tous connus à une précision $N \in \mathbb{N}^*$, avec N strictement plus grand que $\text{cond}_{\leq, \leq_2}$, l'Algorithme FGLM stabilisé 8.1.1 termine et retourne une base de Gröbner G_2 de I pour \leq_2 . Les coefficients des polynômes de G_2 sont connus à précision $N - 2\text{cond}_{\leq, \leq_2}$.*

La preuve de ce théorème est décrite dans la Sous-Section suivante.

8.1.2. Correction, terminaison, précision et complexité

Nous allons montrer le Théorème 8.1.6 qui prouve l'Algorithme 8.1.1. Pour cela, nous commençons par un lemme pour contrôler le comportement du conditionnement de \mathbf{v} au cours de l'algorithme, puis l'appliquons pour montrer chacune des composantes de la preuve de cette algorithme les unes après les autres.

Une remarque préliminaire est néanmoins à faire : en précision infinie, correction et terminaison de l'Algorithme 8.1.1 sont évidentes. En effet, il s'agit simplement de l'Algorithme 6.2.38 avec les tests d'appartenance linéaire et les résolutions de systèmes effectués grâce à la forme normale de Smith au lieu d'un échelonnement (réduit) en lignes.

Un résultat de croissance pour la forme normale de Smith

Afin de contrôler le comportement du conditionnement de \mathbf{v} au cours de l'algorithme et ainsi, suivre la précision, nous utilisons le lemme suivant :

Lemme 8.1.7. *Soit $M \in M_{s, \delta}(K)$ une matrice, avec $s < \delta$ des entiers. Soit $v \in K^\delta$ un vecteur et $M' \in M_{s+1, \delta}(K)$ la matrice obtenue en ajoutant comme $(s+1)$ -ème colonne à M le vecteur v . Soit c le maximum des valuations des facteurs invariants de M , et c' celui correspondant pour M' . Nous supposons que $c, c' \neq +\infty$ (autrement dit, les matrices sont de rang plein). Alors $c \leq c'$.*

Démonstration. Nous utilisons le fait suivant : soit d'_s la plus petite valuation atteinte par un mineur $s \times s$ de M' , et d'_{s+1} la plus petite valuation atteinte par un mineur $(s+1) \times (s+1)$ de M' , alors $c' = d'_{s+1} - d'_s$. Ceci est une conséquence directe du fait que pour un idéal I dans un anneau de valuation discrète V , tout élément de I qui atteint $\min(\text{val}(I))$ engendre I , et réciproquement, et peut être prouvé de manière similaire à la Proposition 1.2.7.

Maintenant, dans le cas qui nous intéresse, soit P, Q, Δ tels que Δ soit la forme normale de Smith de M , $P \in GL_\delta(R)$, $Q \in GL_s(R)$ et $PMQ = \Delta$. Alors, quitte à augmenter trivialement Q en Q' avec $Q'_{s+1, s+1} = 1$, nous pouvons écrire :

8. Stabilité de FGLM

$$PM'Q' = \begin{bmatrix} \pi^{a_1} & & & & w_1 \\ & \ddots & & & \vdots \\ & & 0 & & \vdots \\ & & & \ddots & \vdots \\ & & & & \pi^{a_s} \\ & & & & & \ddots \\ & & & & & & w_\delta \\ & & & & & & & \ddots \\ & & & & & & & & 0 \\ & & & & & & & & & \ddots \\ & & & & & & & & & & w_\delta \end{bmatrix}.$$

Avec cette écriture, $c = a_s$.

De plus, nous pouvons déduire de cette écriture de $PM'Q'$ que d'_{s+1} est de la forme $a_1 + \dots + a_s + \text{val}(w_k)$ pour un certain $k > s$. En effet, les mineurs $(s+1) \times (s+1)$ non nuls de $PM'Q'$ sont tous de cette forme : ils correspondent au choix de $(s+1)$ lignes linéairement indépendantes, et toutes les lignes d'indices au moins $(s+1)$ sont deux à deux liées. Avec ces choix de lignes, le mineur correspondant est le déterminant d'une matrice triangulaire, dont les coefficients diagonaux sont $\pi^{a_1}, \dots, \pi^{a_s}, w_k$.

Par ailleurs, $a_1 + \dots + a_{s-1} + \text{val}(w_k)$ est la valuation d'un mineur $s \times s$ de $PM'Q'$. Par définition, on a alors $d'_s \leq a_1 + \dots + a_{s-1} + \text{val}(w_k)$. Comme $d'_{s+1} = a_1 + \dots + a_s + \text{val}(w_k)$ et $c' = d'_{s+1} - d'_s$, nous en déduisons que $c' \geq a_s = c$, ce que nous souhaitons démontrer. \square

Remarque 8.1.8. Le résultat peut s'étendre au cas où les matrices ne sont pas nécessairement de rang plein, mais dans ce cas, il n'est plus forcément très intéressant. En effet, on a alors $c' = +\infty$ et nécessairement $c \leq c'$ (que M soit de rang plein ou non).

Introduisons maintenant la notation suivante :

Définition 8.1.9. Soit E un R -module et $X \subset E$ un ensemble fini. Nous notons $\text{Vect}_R(X)$ le R -module engendré par les vecteurs de X .

Alors, le lemme précédent a la conséquence suivante :

Lemme 8.1.10. Soit $I, G_1, \leq, \leq_2, B_{\leq}, B_{\leq_2}$ comme dans l'énoncé du théorème 8.1.6. Soit $x^\beta \in \mathcal{B}_{\leq_2}(I)$. Alors : $NF_{\leq}(x^\beta) \in \pi^{-\text{cond}_{\leq, \leq_2}(I)} \text{Vect}_R(\{NF_{\leq}(x^\alpha) | x^\alpha \in B_{\leq_2}, x^\alpha < x^\beta\})$.

Démonstration. La preuve de correction de l'algorithme FGLM (6.2.36 ou 6.2.38) montre que, si \mathbf{v} est la matrice dont les colonnes sont les $NF_{\leq}(x^\alpha)$ avec $x^\alpha \in B_{\leq_2}$ et $x^\alpha < x^\beta$ (écrits dans la base B_{\leq}), alors $NF_{\leq}(x^\beta) \in \text{Im}(\mathbf{v})$.

En appliquant la démonstration de la Proposition 1.3.15, nous obtenons que $NF_{\leq}(x^\beta) \in \pi^{-\text{cond}(\mathbf{v})} \text{Vect}_R(\{NF_{\leq}(x^\alpha) | x^\alpha \in B_{\leq_2}, x^\alpha < x^\beta\})$.

Enfin, le Lemme 8.1.7 nous dit que $\text{cond}(\mathbf{v}) \leq \text{cond}_{\leq, \leq_2}(I)$. Le résultat est alors clair. \square

Correction et terminaison

Nous pouvons maintenant montrer la correction et la terminaison de l'Algorithme 8.1.1 sous réserve d'avoir une précision suffisante en entrée.

Proposition 8.1.11. Soit $G_1, \leq, \leq_2, B_{\leq}, B_{\leq_2}$ comme dans l'énoncé du théorème 8.1.6. Alors en supposant que les coefficients des polynômes de G_1 sont tous connus à une précision $O(\pi^N)$ pour un certain $N \in \mathbb{N}^*$ assez grand, l'Algorithme FGLM stabilisé 8.1.1 termine et retourne une base de Gröbner G_2 de I pour \leq_2 .

Démonstration. Notons tout d'abord qu'il n'y a aucun problème, ni perte de précision lors du calcul des matrices de multiplication : elles sont connues à précision $O(\pi^N)$.

Soit M la matrice dont les colonnes sont les $NF_{\leq}(x^\beta)$ pour $x^\beta \in B_{\leq_2}$. Soit $\text{cond}_{\leq, \leq_2}$ la plus grande valuation d'un facteur invariant de M (le conditionnement de I pour le changement d'ordre de \leq vers \leq_2).

Pour montrer le résultat, nous allons utiliser l'invariant de boucle suivant : lors du début de chaque passage dans la boucle **tant que** de l'Algorithme 8.1.1, nous avons (i), $B_2 \subset B_{\leq_2}$ et (ii) si $x^\beta = B_2[j]x_i$ (où $(j, i) = m$, m pris au début de la boucle), alors tout monôme $x^\alpha <_2 x^\beta$ satisfait

$x^\alpha \in B_{\leq_2}$ ou $NF_{\leq}(x^\alpha) \in \pi^{N-\text{cond}_{\leq, \leq_2}} \text{Vect}_R(NF_{\leq}(B_{\leq_2})) + O(\pi^{N-\text{cond}_{\leq, \leq_2}})$. Ici, $O(\pi^{N-\text{cond}_{\leq, \leq_2}})$ est le R -module engendré par les $\pi^{N-\text{cond}_{\leq, \leq_2} \epsilon}$ pour $\epsilon \in B_{\leq}$.

Nous commençons par montrer que cette proposition constitue bien un invariant de boucle. Elle est bien vérifiée lors du premier passage car $1 \in B_{\leq_2}$. En effet, I est supposé zéro-dimensionnel.

Montrons que cet invariant est bien stable par passage dans la boucle. Soit $x^\beta = B_2[j]x_i$ où $(j, i) = m$. Par construction, x^β est dans le bord de B_2 (i.e. multiple non-trivial d'un monôme de B_2). Comme $B_2 \subset B_{\leq_2}$, nous en déduisons que x^β est soit dans B_{\leq_2} , soit dans le bord de B_{\leq_2} , aussi noté $\mathcal{B}_{\leq_2}(I)$.

Regardons d'abord le second cas. Nous avons alors, par le Lemme 8.1.10,

$$NF_{\leq}(x^\beta) \in \pi^{-\text{cond}_{\leq, \leq_2}} \text{Vect}_R(\{NF_{\leq}(x^\alpha) | x^\alpha \in B_{\leq_2}, x^\alpha < x^\beta\}).$$

À précision finie, ceci nous indique que $\lambda = P_1 v = P_1 NF_{\leq}(x^\beta)$ n'a que des coefficients s'écrivant sous la forme $O(\pi^{\ell'})$ pour ses coefficients sur les lignes $i > s$. Ceci correspond à l'appartenance à l'image de Δ .

Ainsi, le test **si** est passé, et x^β n'est pas ajouté à B_2 . Les points (i) et (ii) restent vérifiés.

Regardons maintenant le premier cas, qui est de $x^\beta \in B_{\leq_2}$. Là encore, nous avons deux possibilités. La première est la suivante : nous avons assez de précision pour que lorsqu'on calcule $\lambda = P_1 v$ où $v = NF_{\leq}(x^\beta)$, nous puissions prouver la non-appartenance de v à $\text{Vect}(NF_{\leq}(B_{\leq_2}))$. Autrement dit, nous sommes alors dans le cas du *sinon*, et x^β est justement ajouté à B_2 . Les points (i) et (ii) restent vérifiés. Il reste alors l'autre possibilité : nous avons pas assez de précision pour que lorsqu'on calcule $\lambda = P_1 v$ où $v = NF_{\leq}(x^\beta)$, nous puissions prouver la non appartenance de v à $\text{Vect}(B_{\leq_2})$. Autrement dit, nous obtenons numériquement que $NF_{\leq}(x^\beta) \in \pi^{-\text{cond}(\mathbf{v})} \text{Vect}_R(\{NF_{\leq}(x^\alpha) | x^\alpha \in B_{\leq_2}, x^\alpha < x^\beta\}) + O(\pi^{N-\text{cond}(\mathbf{v})})$. Dans ce cas, nous passons le test **si**, et, comme $\text{cond}(\mathbf{v}) \leq \text{cond}_{\leq, \leq_2}$, les points (i) et (ii) restent vérifiés.

Cet invariant de boucle est maintenant suffisant pour clore la démonstration. En effet, du fait d'avoir toujours $B_2 \subset B_{\leq_2}$, nous en déduisons que L est toujours inclus dans $B_{\leq_2} \cup \mathcal{B}_{\leq_2}(I)$, et comme un monôme ne peut être traité plusieurs fois dans la boucle **tant que**, il y a au plus $n\delta$ passages dans cette boucle. D'où la terminaison.

Concernant la correction, si nous passons le test **si** avec $\text{card}(B_2) = \delta = \text{card}(B_{\leq_2})$, alors, du fait de l'inclusion déjà montrée, nous avons l'égalité $B_2 = B_{\leq_2}$. Dans ce cas, les monômes de têtes qui ont passé le test **si** sont alors nécessairement dans le bord $\mathcal{B}_{\leq_2}(I)$, et s'écrivent donc bien dans A/I en fonction des monômes de B_2 plus petits. En d'autres termes, la résolution de système linéaire avec l'hypothèse d'appartenance linéaire fournit bien un polynôme de I . In fine, G_2 est bien une base de Gröbner de I pour \leq_2 .

Dans le second cas, celui où nous échouons au test **si** en ayant $\text{card}(B_2) \neq \delta$, la précision n'a pas été suffisante.

Le résultat est donc prouvé. \square

Analyse de la perte de précision

Nous pouvons maintenant analyser la perte de précision lors de l'exécution de l'Algorithme FGLM stabilisé 8.1.1, et ce faisant, nous estimons la précision nécessaire en entrée pour que son exécution se passe sans erreur. Pour cela, nous utilisons la notion de conditionnement donnée par la Définition 8.1.4, et montrons qu'elle contrôle bien le comportement de la précision lors de l'exécution de l'Algorithme FGLM stabilisé 8.1.1. C'est ce que montre la proposition suivante :

Proposition 8.1.12. *Soit $I, G_1, \leq, \leq_2, B_{\leq}, B_{\leq_2}$ comme dans l'énoncé du théorème 8.1.6. Soit M la matrice dont les colonnes sont les $NF_{\leq}(x^\beta)$ pour $x^\beta \in B_{\leq_2}$. Alors, si les coefficients des polynômes de G_1 sont tous connus à une précision $N \in \mathbb{N}^*$, avec N strictement plus grand que $\text{cond}_{\leq, \leq_2}$, l'Algorithme FGLM stabilisé 8.1.1 termine et retourne une base de Gröbner approchée G_2 de I pour \leq_2 . Les coefficients des polynômes de G_2 sont connus à précision $N - 2\text{cond}_{\leq, \leq_2}$.*

Démonstration. Lors de l'exécution de l'Algorithme 8.1.1, le moment où la précision utilisée entre en compte est lors de la résolution du système linéaire, lorsque le nouveau vecteur v est dans $\text{Im}(\mathbf{v})$. Supposons que les coefficients des polynômes de G_1 soient tous connus à la même précision N , alors il en est de même des T_i et ainsi, de \mathbf{v} et sa forme normale de Smith approchée. En conséquence, les

8. Stabilité de FGLM

coefficients de v sont aussi connus à précision N . D'après la Proposition 1.3.15, et avec l'hypothèse d'appartenance de v à $Im(\mathbf{v})$, il suffit alors d'une précision N strictement supérieure à la plus grande valuation c d'un facteur invariant de \mathbf{v} pour pouvoir résoudre le système linéaire $\mathbf{v}W = v$, et les coefficients de W sont déterminés à précision $N - 2c$. Le lemme 8.1.7, nous permet alors de conclure qu'à tout moment, $c \leq cond_{\leq, \leq 2}$, d'où le résultat. \square

Un exemple

Exemple 8.1.13. Nous appliquons l'Algorithme FGLM stabilisé 8.1.1 sur la famille de polynôme de l'exemple 7.2.15.

Celle-ci est donné par $F = (f_1, f_2, f_3) \in \mathbb{Q}_2[x, y, z]$ avec les f_i connus à précision $O(2^{10})$: $f_1 = (2 + O(2^{10}))x + (1 + O(2^{10}))z$, $f_2 = (1 + O(2^{10}))x^2 + (1 + O(2^{10}))y^2 - (2 + O(2^{10}))z^2$ et $f_3 = (4 + O(2^{10}))y^2 + (1 + O(2^{10}))yz + (8 + O(2^{10}))z^2$. Nous prenons $D = 3$, la borne de Macaulay.

Nous calculons d'abord une base de Gröbner réduite pour grevlex avec l'Algorithme 7.2.7, et obtenons :

$$(x + (2^{-1} + O(2^8))z, y^2 + (2^{-2} + 2 + 2^2 + 2^3 + 2^4 + 2^5 + O(2^6))z^2, yz + (1 + 2 + 2^2 + 2^3 + O(2^8))z^2, z^3).$$

L'escalier est alors $[1, z, y, z^2]$. Nous appliquons maintenant l'Algorithme FGLM stabilisé 8.1.1 à partir de cette base de Gröbner réduite pour en déduire une pour l'ordre lexicographique avec $z > y > x$. Nous obtenons la matrice suivante comme forme normale de Smith de la matrice de changement de base :

$$\begin{bmatrix} 2^{-2} & 0 & 0 & 0 \\ 0 & 2^{-1} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Le nouvel escalier est $[1, x, x^2, y]$. Nous obtenons alors comme base de Gröbner (rendue minimale) en sortie :

$$(x^3, xy + (2 + 2^5 + 2^6 + 2^7 + 2^8 + O(2^9))x^2, y^2 + (1 + 2^3 + 2^4 + 2^5 + 2^6 + 2^7 + O(2^8))x^2, z + (2 + O(2^{10}))x).$$

Complexité

Pour conclure la preuve du Théorème 8.1.6, il nous reste à estimer la complexité de l'Algorithme 8.1.1. Pour ce qui est du calcul des matrices de multiplication, il n'y a pas de modification du point de vue de la complexité, et l'essentiel de ce que nous devons prouver dépend du calcul de la forme normale de Smith itérée. Son comportement est étudié dans le lemme suivant :

Lemme 8.1.14. *Soit $1 \leq s \leq \delta$ et $prec$ des entiers, $k \in \llbracket 1, s \rrbracket$ et $M, C^{(k)}$ deux matrices dans $M_{\delta \times s}(K)$. Supposons que les coefficients de M vérifient $M_{i,j} = m_{i,j}\delta_{i,j} + O(\pi^{prec})$ pour des $m_{i,j} \in K$ et que les coefficients de $C^{(k)}$ vérifient $C_{i,j}^{(k)} = c_{i,j}\delta_{j,k} + O(\pi^{prec})$ pour des $c_{i,j} \in K$. Soit $C_{FNS}(M + C^{(k)})$ le nombre d'opérations sur les lignes et colonnes pour calculer la forme normale de Smith approchée de $M + C^{(k)}$ à précision $O(\pi^{prec})$. Alors $C_{FNS}(M + C^{(k)}) \leq s\delta$.*

Démonstration. Nous montrons ce résultat par récurrence sur s . Pour $s = 1$, pour tout $\delta, prec, k, M$ et $C^{(k)}$, le résultat est clair.

Supposons que pour un certain $s \in \mathbb{N}^*$, on ait pour tout $\delta, prec, k$, et M et $C^{(k)} \in M_{\delta \times (s-1)}(K)$ comme dans l'énoncé, on ait $C_{FNS}(M + C^{(k)}) \leq (s-1)\delta$.

Alors, soit $\delta \geq s$ et $k \in \llbracket 1, s \rrbracket$ et $rec \in \mathbb{N}$. Soit $M, C^{(k)}$ deux matrices dans $M_{\delta \times s}(K)$ tels que leurs coefficients vérifient $M_{i,j} = m_{i,j}\delta_{i,j} + O(\pi^{prec})$, pour des $m_{i,j} \in K$, et $C_{i,j}^{(k)} = c_{i,j}\delta_{j,k} + O(\pi^{prec})$ pour des $c_{i,j} \in K$. Soit $N = M + C^{(k)}$.

Appliquons l'Algorithme 1.3.5 jusqu'avant l'appel récursif. Supposons que le coefficient utilisé comme pivot, le coefficient $N_{i,j}$ qui atteint le min des $val(N_{i,j})$, est $N_{1,1}$. Alors 1 opération sur les colonnes est effectuée lors du parcours des deux boucles **pour** consécutives de l'Algorithme 1.3.5. Le seul autre cas est que ce pivot est un $N_{i,k}$ pour un certain i . Alors $\delta - 1$ opérations sur les lignes et 1 opération sur les colonnes sont effectuées.

La matrice $N' = \tilde{N}_{i \geq 2, j \geq 2}$ peut alors s'écrire $N' = M' + C'^{(k)}$ avec M' et $C'^{(k)}$ dans $M_{(\delta-1) \times (s-1)}(K)$ de la forme voulue, pour $k = s - 1$ si le pivot $N_{i,j}$ est $N_{1,1}$ et $k = i$ si c'est $N_{i,s}$. En appliquant l'hypothèse de récurrence sur N' , nous obtenons bien $C_{FNS}(M + C^{(k)}) \leq \delta + (\delta - 1) \times (s - 1) \leq \delta s$.

Le résultat est donc prouvé par récurrence. \square

Nous avons donc le résultat suivant concernant la complexité de l'Algorithme 8.1.1 :

Proposition 8.1.15. *Soient une base de Gröbner approchée réduite G_1 pour un ordre monomial \leq d'un idéal $I \subset A$ de dimension zéro et de degré δ , et un ordre monomial \leq_2 . Supposons que les coefficients de G_1 sont connus à précision $O(\pi^N)$ pour un certain $N > \text{cond}_{\leq, \leq_2}$. Alors, la complexité de l'exécution de l'Algorithme 8.1.1 est en $O(n\delta^3)$ opérations dans K à précision absolue $O(\pi^N)$.*

Démonstration. Tout d'abord, remarquons que le calcul des matrices de multiplication est en $O(n\delta^3)$ opérations à précision $O(\pi^N)$. Maintenant, considérons l'intérieur de la boucle **tant que** de l'Algorithme 8.1.1. Les calculs de forme normale de Smith approchée, par la procédure Update, sont en $O(\delta^2)$ opérations à précision $O(\pi^N)$ grâce au Lemme 8.1.14. Les résolutions de système linéaire grâce à la Proposition 1.3.15 sont elles aussi en $O(\delta^2)$ opérations à précision $O(\pi^N)$. Il y a au plus $n\delta$ passages dans cette boucle du fait de la preuve de terminaison en Proposition 8.1.11. Le résultat est donc prouvé. \square

Nous rappelons que la complexité de l'algorithme FGLM dans le cas classique proposé par l'Algorithme 6.2.38 est elle aussi en $O(n\delta^3)$ opérations sur le corps de base.

8.2. Cas d'un idéal en position générale

Dans cette Section, nous analysons le cas particulier de l'application de l'algorithme FGLM dans sa principale variante, celle pour le calcul d'une base de Gröbner pour l'ordre lex pour un idéal en position générale. Nous verrons que la perte de précision s'exprime à nouveau naturellement grâce au calcul de la forme normale de Smith, et que les gains de complexité obtenus à précision infinie lors de l'étude dans la Sous-Section 6.2.2 demeurent.

8.2.1. Présentation de l'algorithme

Tout d'abord, nous introduisons, similairement aux Algorithmes 6.2.48 et 6.2.53, deux versions pour l'algorithme FGLM pour le calcul d'une base de Gröbner pour un idéal en position générale, selon que l'ordre monomial initial soit grevlex ou non.

Une première version

Nous commençons par un algorithme FGLM pour le calcul d'une base de Gröbner pour un idéal en position générale à partir d'une base de Gröbner pour un ordre monomial donné. Il s'agit essentiellement d'adapter l'Algorithme 6.2.48 en effectuant les résolutions de systèmes linéaires

8. Stabilité de FGLM

grâce au calcul d'une seule forme normale de Smith.

Algorithme 8.2.1 : Algorithme FGLM stabilisé pour un idéal en position générale

entrée : Une base de Gröbner approchée réduite G_1 pour un ordre monomial \leq d'un idéal $I \subset A$ de dimension zéro et de degré δ , $B_{\leq} = (1 = \epsilon_1 \leq \epsilon_2 \leq \dots \leq \epsilon_{\delta})$ la base canonique de A/I pour \leq .
 I est en position générale.
sortie : Une base de Gröbner approchée G_2 de I pour \leq_{lex} .

début

Calculer les matrices de multiplications T_1, \dots, T_n pour I et \leq avec l'Algorithme 6.2.34 ;
 $G_2 := \emptyset$;
pour i de 1 à $n - 1$ **faire**
 Calculer $\mathbf{y}[i] := T_i 1$;
 $\mathbf{z}[0] := 1$;
pour i de 1 à δ **faire**
 Calculer $\mathbf{z}[i] = T_n \mathbf{z}[i - 1]$;
 $M := \text{Mat}_{B_{\leq}}(\mathbf{z}[0], \dots, \mathbf{z}[\delta - 1])$;
Calculer Δ la forme normale de Smith de M avec $\Delta = PMQ$;
si le rang de M est bien δ **alors**
 pour i de 1 à $n - 1$ **faire**
 Trouver U tel que $\mathbf{y}[i] = -M \cdot U$ grâce à P, Q, Δ et le Théorème 1.3.14 ;
 $h_i(T) := \sum_{i=0}^{\delta-1} U[i] T^i$;
 Trouver U tel que $\mathbf{z}[\delta] = -M \cdot U$ grâce à P, Q, Δ et le Théorème 1.3.14 ;
 $h_n(T) := T^{\delta} + \sum_{i=0}^{\delta-1} U[i] T^i$;
 Retourner $x_1 - h_1(x_n), \dots, x_{n-1} - h_{n-1}(x_n), h_n(x_n)$;
sinon
 Retourner "Erreur, la précision est insuffisante"

Cas de grevlex comme ordre monomial initial

Dans le cas particulier où l'ordre monomial initial est grevlex et que l'on souhaite calculer par l'algorithme FGLM une base de Gröbner pour lex d'un idéal en position générale, nous pouvons adapter l'Algorithme 6.2.53. Il s'agit de même d'effectuer les résolutions de systèmes linéaires en

utilisant un calcul de forme normale de Smith.

Algorithme 8.2.2 : Algorithme FGLM stabilisé pour un idéal en position générale à partir de grevlex

entrée : Une base de Gröbner approchée réduite G_1 pour l'ordre grevlex d'un idéal $I \subset A$ de dimension zéro et de degré δ , $B_{\leq} = (1 = \epsilon_1 \leq \epsilon_2 \leq \dots \leq \epsilon_\delta)$ la base canonique de A/I pour \leq .
On exige que I satisfasse la Proposition 6.2.51.
 I est en position générale.

sortie : Une base de Gröbner approchée G_2 de I pour \leq_{lex} .

début

Calculer la matrice de multiplication T_n pour I et grevlex à partir de G (Prop 6.2.52) ;
 $G_2 := \emptyset$;
pour i de 1 à $n - 1$ **faire**
 Grâce à la Prop. 6.2.50 calculer $\mathbf{y}[i] := T_i 1$ à partir de G ;
 $\mathbf{z}[0] := 1$;
pour i de 1 à δ **faire**
 Calculer $\mathbf{z}[i] = T_n \mathbf{z}[i - 1]$;
 $M := \text{Mat}_{B_{\leq}}(\mathbf{z}[0], \dots, \mathbf{z}[\delta - 1])$;
Calculer Δ la forme normale de Smith de M avec $\Delta = PMQ$;
si le rang de M est bien δ **alors**
 pour i de 1 à $n - 1$ **faire**
 Trouver U tel que $\mathbf{y}[i] = -M \cdot U$ grâce à P, Q, Δ et le Théorème 1.3.14 ;
 $h_i(T) := \sum_{j=1}^{\delta-1} U[j] T^j$;
 Trouver U tel que $\mathbf{z}[\delta] = -M \cdot U$ grâce à P, Q, Δ et le Théorème 1.3.14 ;
 $h_n(T) := T^\delta + \sum_{j=1}^{\delta-1} U[j] T^j$;
 Retourner $x_1 - h_1(x_n), \dots, x_{n-1} - h_{n-1}(x_n), h_n(x_n)$;
sinon
 Retourner "Erreur, la précision est insuffisante"

Remarque 8.2.3. Si l'idéal I est faiblement grevlex (ou plus fort encore, si les polynômes donnés en entrée satisfont l'hypothèse **H2** du chapitre précédent), alors la Proposition 6.2.51 est satisfaite.

Énoncé du résultat

Nous pouvons maintenant énoncer le théorème suivant concernant le calcul d'une base de Gröbner par FGLM pour un idéal en position générale :

Théorème 8.2.4. Soit G_1 une base de Gröbner approchée réduite pour un ordre monomial \leq d'un idéal $I \subset A$ de dimension zéro et de degré δ . Soit B_{\leq} la base canonique de A/I pour \leq . Supposons que les coefficients des polynômes de G_1 soient tous connus à précision $O(\pi^N)$ pour un certain $N \in \mathbb{N}^*$, hormis les coefficients de têtes qui valent exactement 1. Notons $m = \text{cond}_{\leq, lex}(I)$ et supposons que $m < N$. Supposons enfin que I est en position générale. Alors l'Algorithme 8.2.1 calcule une représentation univariée de I . De plus, ses coefficients sont connus à précision $O(\pi^{N-2m})$. La complexité est en $O(n\delta^3)$ opérations dans K (à cause du calcul des matrices de multiplication). Si \leq est grevlex et que l'on utilise l'Algorithme 8.2.2, nous avons le même résultat, pour une complexité en $O(\delta^3) + O(n\delta^2)$.

La Sous-Section suivante est consacrée à prouver ce théorème.

8.2.2. Correction, terminaison et précision

Nous pouvons maintenant étudier et prouver ces deux algorithmes. Nous montrons d'abord leur correction et leur terminaison :

Proposition 8.2.5. Supposons que les coefficients des polynômes de la base de Gröbner approchée réduite G_1 soient connus à précision suffisamment grande, et que l'idéal $I = \langle G \rangle$ est en position

8. Stabilité de FGLM

générale. Alors l'Algorithme 8.2.1 (et l'Algorithme 8.2.2 dans le cas où l'ordre monomial initial est grevlex) termine et renvoie une base de Gröbner pour l'ordre lexicographique de I , fournissant une représentation univariée. Les temps des calcul sont identiques aux cas de précision infinie.

Démonstration. Lorsqu'on peut certifier que le rang de M est bien δ , la dimension de A/I , alors nous pouvons certifier que I admet bien une base de Gröbner fournissant une représentation univariée. Correction et terminaison sont alors claires avec les preuves des corrections et terminaisons des Algorithmes 6.2.48 et 6.2.53. \square

Il reste alors à analyser la perte de précision. Pour cela nous utilisons de nouveau le conditionnement de I pour le changement d'ordre de \leq vers lex :

Proposition 8.2.6. *Soit G_1 une base de Gröbner approchée réduite d'un idéal de dimension zéro et de degré δ $I \subset A$, pour un ordre monomial \leq . Soit B_{\leq} la base canonique de A/I pour \leq . Supposons que les coefficients des polynômes de G_1 soient tous connus à précision $O(\pi^N)$ pour un certain $N \in \mathbb{N}^*$, hormis les coefficients de têtes qui valent exactement 1. Notons $m = \text{cond}_{\leq, \text{lex}}(I)$. Supposons que $m < N$. Supposons enfin que I est en position générale. Alors l'Algorithme 8.2.1 calcule une base de Gröbner pour lex qui fournit une représentation univariée de I . De plus, ses coefficients sont connus à précision $O(\pi^{N-2m})$. Le résultat est identique si \leq est grevlex et que l'on utilise l'Algorithme 8.2.2.*

Démonstration. Il n'y a pas de perte de précision lors des calculs des matrices de multiplication, que ce soit pour l'Algorithme 8.2.1 ou l'Algorithme 8.2.2, vu que seuls additions et multiplication de nombres connus à précision $O(\pi^l)$ sont effectués. Le calcul de la matrice $M := \text{Mat}_{B_{\leq}}(NF_{\leq}(1), \dots, NF_{\leq}(x_n^{\delta-1}))$ se fait de même sans perte de précision. Ensuite, les Algorithmes 8.2.1 et 8.2.2 ne font que résoudre des systèmes linéaires par la matrice M . Le résultat est alors direct avec le Théorème 1.3.14. \square

8.3. Implémentation

Une implémentation jouet en Sage [S⁺11] des algorithmes précédents est disponible sur <http://perso.univ-rennes1.fr/tristan.vaccon/fglm.sage>. Comme le but de cette implémentation est l'étude de la précision, elle n'est pas nécessairement optimisée pour ce qui est du temps de calcul. Nous avons appliqué l'algorithme F5-Matriciel à des polynômes homogènes de degrés donnés, ou son extension au cas affine, et avec des coefficients pris aléatoirement dans \mathbb{Z}_p (pour la mesure de Haar) : f_1, \dots, f_s , de degrés d_1, \dots, d_s dans $\mathbb{Z}_p[X_1, \dots, X_s]$, connus à précision $O(p^{\text{prec}})$, et pour l'ordre grevlex, avec D la borne de Macaulay. Nous avons ensuite appliqué un algorithme FGLM, rapide ou non, sur les bases de Gröbner obtenues pour en déduire une base de Gröbner pour l'ordre lexicographique.

Cette expérience est réalisé nb_{test} fois pour chaque choix de paramètres et sont notés la perte de précision maximale (hors échec), moyenne (hors échec), et le nombre d'échecs. Ce dernier apparaît comme un couple où la première composante est le nombre d'échecs pour la partie F5-Matriciel et la seconde composante pour la partie FGLM. Les résultats sont consignés dans les tableaux suivant :

| $d =$ | nb_{test} | affine | fast | D | p | $prec$ | perte maximale | perte moyenne | échecs |
|---------|-------------|--------|------|---|-------|--------|----------------|---------------|--------|
| [2,3,3] | 50 | non | non | 6 | 2 | 150 | 27 | 1,8 | (0,0) |
| [2,3,3] | 50 | non | non | 6 | 7 | 150 | 6 | 0,4 | (0,0) |
| [2,3,3] | 50 | oui | non | 6 | 2 | 150 | 142 | 50 | (0,2) |
| [2,3,3] | 50 | oui | non | 6 | 7 | 150 | 71 | 12 | (0,0) |
| [2,3,3] | 50 | oui | oui | 6 | 2 | 150 | 148 | 48 | (0,1) |
| [2,3,3] | 50 | oui | oui | 6 | 7 | 150 | 79 | 13 | (0,0) |
| [3,3,3] | 20 | non | non | 7 | 2 | 150 | 21 | 3 | (0,0) |
| [3,3,3] | 20 | non | non | 7 | 7 | 150 | 11 | 0,9 | (0,0) |
| [3,3,3] | 20 | oui | non | 7 | 2 | 150 | 150 | 78 | (0,0) |
| [3,3,3] | 20 | oui | non | 7 | 7 | 150 | 150 | 78 | (0,0) |
| [3,3,3] | 20 | oui | oui | 7 | 2 | 150 | 145 | 65 | (0,1) |
| [3,3,3] | 20 | oui | oui | 7 | 7 | 150 | 100 | 27 | (0,0) |
| [3,3,4] | 20 | non | non | 8 | 2 | 150 | 21 | 3 | (0,0) |
| [3,3,4] | 20 | non | non | 8 | 7 | 150 | 5 | 0,5 | (0,0) |
| [3,3,4] | 20 | oui | non | 8 | 2 | 150 | 149 | 92 | (0,5) |
| [3,3,4] | 20 | oui | non | 8 | 7 | 150 | 130 | 36 | (0,0) |
| [3,3,4] | 20 | oui | oui | 8 | 2 | 150 | 150 | 89 | (0,7) |
| [3,3,4] | 20 | oui | oui | 8 | 7 | 150 | 88 | 22 | (0,0) |
| $d =$ | nb_{test} | affine | fast | D | p | $prec$ | perte maximale | perte moyenne | échecs |
| [2,2,2] | 50 | non | non | 4 | 2 | 150 | 18 | 1 | (0,0) |
| [2,2,2] | 50 | non | non | 4 | 7 | 150 | 6 | 0,4 | (0,0) |
| [2,2,2] | 50 | non | non | 4 | 65519 | 150 | 0 | 0 | (0,0) |
| [2,2,2] | 50 | oui | non | 4 | 2 | 150 | 66 | 16 | (0,0) |
| [2,2,2] | 50 | oui | non | 4 | 7 | 150 | 21 | 4,5 | (0,0) |
| [2,2,2] | 50 | oui | non | 4 | 65519 | 150 | 0 | 0 | (0,0) |
| [2,2,2] | 50 | oui | oui | 4 | 2 | 150 | 42 | 14 | (0,0) |
| [2,2,2] | 50 | oui | oui | 4 | 7 | 150 | 17 | 2,7 | (0,0) |
| [2,2,2] | 50 | oui | oui | 4 | 65519 | 150 | 0 | 0 | (0,0) |
| [3,3,3] | 20 | non | non | 7 | 2 | 150 | 17 | 2,4 | (0,0) |
| [3,3,3] | 20 | non | non | 7 | 7 | 150 | 6 | 0,8 | (0,0) |
| [3,3,3] | 20 | non | non | 7 | 65519 | 150 | 0 | 0 | (0,0) |
| [3,3,3] | 20 | oui | non | 7 | 2 | 150 | 146 | 64 | (0,1) |
| [3,3,3] | 20 | oui | non | 7 | 7 | 150 | 70 | 18 | (0,0) |
| [3,3,3] | 20 | oui | non | 7 | 65519 | 150 | 0 | 0 | (0,0) |
| [3,3,3] | 20 | oui | oui | 7 | 2 | 150 | 160 | 70 | (0,0) |
| [3,3,3] | 20 | oui | oui | 7 | 7 | 150 | 121 | 25 | (0,0) |
| [3,3,3] | 20 | oui | oui | 7 | 65519 | 150 | 0 | 0 | (0,0) |
| [4,4,4] | 20 | non | non | 7 | 2 | 150 | 28 | 5,2 | (0,0) |
| [4,4,4] | 20 | non | non | 7 | 7 | 150 | 8 | 1 | (0,0) |
| [4,4,4] | 20 | non | non | 7 | 65519 | 150 | 0 | 0 | (0,0) |
| [4,4,4] | 20 | oui | non | 7 | 2 | 150 | 150 | 118 | (0,11) |
| [4,4,4] | 20 | oui | non | 7 | 7 | 150 | 149 | 62 | (0,1) |
| [4,4,4] | 20 | oui | non | 7 | 65519 | 150 | 0 | 0 | (0,0) |
| [4,4,4] | 20 | oui | oui | 7 | 2 | 150 | 156 | 124 | (0,15) |
| [4,4,4] | 20 | oui | oui | 7 | 7 | 150 | 129 | 47 | (0,2) |
| [4,4,4] | 20 | oui | oui | 7 | 65519 | 150 | 0 | 0 | (0,0) |

Nous pouvons remarquer que ces résultats suggèrent une différence d'ordre de grandeur sur la perte de précision entre le cas homogène et le cas affine.¹ Il paraît aussi clair que la perte de précision décroît avec le choix de p : sur des petites instances comme ici, $p = 65519$ rend les pertes de précision très peu probables.

1. Nous laissons à de futurs travaux le soin de donner une explication quantitative à ce fait. Qualitativement, nous pouvons remarquer que, à degrés initiaux donnés, plus de calculs (et en particulier des calculs impliquant une perte de précision) sont effectués dans le cas affine, du fait de l'étape d'inter-réduction.

9. Une approche tropicale

“Show me your moves!”

Captain Falcon, *Super Smash Bros.*

“THAT’S MORTALS FOR YOU,
Death continued. THEY’VE ONLY
GOT A FEW YEARS IN THIS
WORLD AND THEY SPEND
THEM ALL IN MAKING THINGS
COMPLICATED FOR
THEMSELVES. FASCINATING.”

Death, Terry Pratchett, *Mort*

Au cours de ce dernier chapitre, nous nous intéressons au calcul des bases de Gröbner tropicales. Nous débutons par présenter en Section 9.1 nos résultats ainsi que quelques raisons, issues de la géométrie tropicale, de s’intéresser aux bases de Gröbner tropicales. En Section 9.2, nous présentons deux variantes de l’algorithme F5-Matriciel pour calculer de telles bases, puis étudions l’application de cet algorithme sur des CDVF à précision finie. Nous nous intéressons ensuite, avec la Section 9.3, à la description, à l’étude et aux applications d’un algorithme FGLM pour passer d’une base de Gröbner tropicale à une base de Gröbner classique en dimension zéro. La Section 9.4 adapte les idées précédentes au cas d’idéaux de dimension zéro non-homogènes. Enfin, la Section 9.5 présente les résultats numériques obtenus en implémentant les algorithmes précédents.

9.1. Introduction et motivations tropicales

9.1.1. Résultats principaux

Algorithmes F5-Matriciels Soit \mathcal{K} un corps muni d’une valuation val . Soit \geq un ordre sur les termes de $\mathcal{K}[X_1, \dots, X_n]$ comme dans la Définition 9.1.6, défini avec $w \in Im(val)^n$ et un ordre monomial \geq_1 . En suivant [CM13], nous définissons des D -bases de Gröbner tropicales comme pour les bases de Gröbner classiques.

Nous décrivons en l’Algorithme 9.2.3 un algorithme d’échelonnement pour les matrices de Maculay adapté au contexte tropical, puis nous montrons que le critère F5 reste disponible en tropical. Nous en déduisons une description d’un algorithme F5-Matriciel tropicale (Algorithme 9.2.8), adaptation de l’algorithme F5-Matriciel *naïf* muni de l’algorithme d’échelonnement tropical. Nous avons alors le résultat suivant :

Proposition 9.1.1. *Soient $(f_1, \dots, f_s) \in \mathcal{K}[X_1, \dots, X_n]^s$ des polynômes homogènes. Alors, l’algorithme F5-Matriciel tropical calcule une D -base de Gröbner tropicale de $\langle f_1, \dots, f_s \rangle$. Le temps de calcul est en $O\left(s^2 D^{\binom{n+D-1}{D}}\right)$ opérations sur K , lorsque $D \rightarrow +\infty$. Si (f_1, \dots, f_s) est régulière, le temps de calcul est en $O\left(s D^{\binom{n+D-1}{D}}\right)$.*

La borne de Macaulay sur D reste disponible dans le cas tropical. En outre, l’algorithme F5-Matriciel tropical est applicable lorsque les f_i en entrée forment une suite régulière mais que leurs coefficients ne sont connus qu’à précision finie. En effet, si K est un CDVF à précision finie, soit $(f_1, \dots, f_s) \in K[X_1, \dots, X_n]^s$. Nous définissons une borne sur la précision requise, $prec_{MF5trop}((f_1, \dots, f_s), D, \geq)$, et une sur la perte de précision, $loss_{MF5trop}((f_1, \dots, f_s), D, \geq)$,

9. Une approche tropicale

qui dépendent explicitement des matrices de Macaulay définies par les f_i , et telles que nous avons le résultat suivant concernant la stabilité du calcul de bases de Gröbner tropicales :

Proposition 9.1.2. *Soit $F = (f_1, \dots, f_s) \in K[X_1, \dots, X_n]^s$ une suite régulière de polynômes homogènes. Soit (f'_1, \dots, f'_s) des approximations de F avec précision $O(\pi^l)$ sur leurs coefficients, avec $l > \text{prec}_{MF5trop}(F, D, \geq)$. Alors, par l'Algorithme F5-Matriciel tropical, nous pouvons calculer g'_1, \dots, g'_t , base de Gröbner approchée de $\langle F \rangle$ pour \geq à précision $O(\pi^{l - \text{loss}_{MF5trop}(F, D, \geq)})$.*

Ceci contraste avec le cas des bases de Gröbner classiques, pour un ordre monomial ω , sur K que nous avons étudié au Chapitre 7. En effet, l'hypothèse de structure **H2** qui requiert que les idéaux $\langle f_1, \dots, f_i \rangle$ soient faiblement- ω n'est plus nécessaire (voir la Sous-Section 9.2.2). Elle est seulement remplacée par une hypothèse sur la précision initiale qui peut être un peu plus dure que celle que nous avons définie au Chapitre 7. Dans le cas particulier du poids $w = (0, \dots, 0)$, la majoration sur la perte de précision est la meilleure que nous obtenons, et des expériences numériques nous montrent qu'elle semble en moyenne très faible.

Nous montrons aussi qu'une version plus rapide de l'algorithme F5-Matriciel, où l'on utilise des signatures pour construire les matrices de Macaulay degré par degré, peut être adaptée au cas tropical. Pour cela, nous décrivons un échelonnement matriciel, par le calcul de ce que nous nommons une forme LUP, et qui est compatible avec les signatures. Nous en déduisons un algorithme F5-Matriciel avec signatures (voir les Algorithmes 9.2.22 et 9.2.25). Nous obtenons alors le résultat suivant :

Proposition 9.1.3. *Soit $(f_1, \dots, f_s) \in \mathcal{K}[X_1, \dots, X_n]^s$ des polynômes homogènes. Alors l'algorithme F5-Matriciel avec signature calcule une D -base de Gröbner tropicale de $\langle f_1, \dots, f_s \rangle$.*

Le temps de calcul est en $O\left(sD^{(n+D-1)}\right)^3$ opérations sur K , pour $D \rightarrow +\infty$ et $O\left(D^{(n+D-1)}\right)^3$ lorsque les polynômes en entrée forment une suite régulière.

Algorithmes FGLM Une fois vu que les bases de Gröbner tropicales constituaient un objet pouvant se calculer de manière souvent plus stable que les bases de Gröbner classiques, il est intéressant de se demander s'il est possible de calculer une base de Gröbner à partir d'une base de Gröbner tropicale.

Nous montrons avec la Proposition 9.3.11, et après s'être intéressé au calcul des matrices de multiplication, que l'algorithme FGLM s'adapte sans difficulté au cas où la base de départ est tropicale.

Dans le cas de la précision finie, l'estimation de la perte de précision est identique au cas classique. Ceci nous permet avec la Proposition 9.3.14 de majorer la précision requise et la perte de précision pour calculer une base de Gröbner d'une suite régulière en effectuant à la suite un calcul de base de Gröbner tropicale puis une application de l'algorithme FGLM.

Enfin, nous pouvons directement en déduire la version affine du cas précédent, que nous énonçons au Théorème 9.4.1 : pour $F = (f_1, \dots, f_n)$ dans $A = K[X_1, \dots, X_n]$, tel que les composantes homogènes de plus haut degré $F^h = (f_1^h, \dots, f_n^h)$ forment une suite régulière, il est possible de calculer à précision finie une base de Gröbner approchée de $\langle f_1, \dots, f_n \rangle$. Nous majorons la précision requise et la perte de précision par les mineurs des matrices de Macaulay définies par (f_1^h, \dots, f_n^h) et des matrices de changement de base dans $A/\langle F^h \rangle$ et $A/\langle F \rangle$.

9.1.2. Motivations tropicales

Nous présentons dans cette Sous-Section quelques résultats classiques de géométrie algébrique tropicale, ainsi que quelques résultats récents de géométrie algébrique tropicale effective. Ils motiveront en partie notre étude du calcul des bases de Gröbner tropicales.

Pour cette Sous-Section, nous relâchons les hypothèses suivantes sur \mathcal{K} : \mathcal{K} est muni d'une valuation val , mais nous ne demandons pas que la valuation le rende complet ni qu'elle soit discrète. $\Gamma = val(\mathcal{K}^*)$.

Géométrie algébrique tropicale

Variétés tropicales et bases de Gröbner tropicales Soit I un idéal homogène de \mathcal{A} , et $V(I) \subset \mathbb{P}_{\mathcal{K}}^{n-1}$ la variété projective définie par I . Alors la variété tropicale définie par I , ou le tropicalisé de $V(I)$, est $Trop(I) = \overline{val(V(I) \cap (\mathcal{K}^*)^n)}$ (adhérence dans \mathbb{R}^n pour la topologie classique). $Trop(I)$ est un complexe polyédral et peut être vu comme un reflet combinatoire de $V(I)$: de nombreuses propriétés de $V(I)$ peuvent être vues combinatoirement à partir de $Trop(I)$.

Maintenant, si $w \in \Gamma^n$, nous définissons un ordre sur les termes de \mathcal{A} :

Définition 9.1.4. Soit $a, b \in \mathcal{K}$ et x^α, x^β deux monômes de \mathcal{A} , nous notons $ax^\alpha \geq_w bx^\beta$ si $val(a) + w \cdot \alpha \leq val(b) + w \cdot \beta$. Naturellement, il peut arriver que $ax^\alpha \neq bx^\beta$ et $val(a) + w \cdot \alpha = val(b) + w \cdot \beta$.

Pour tout $f \in \mathcal{A}$, nous définissons $LT_{\geq_w}(f)$ comme le polynôme formé des termes atteignant le maximum pour \leq_w sur les termes de f . Nous définissons de même $LT_{\geq_w}(I)$, pour I idéal de \mathcal{A} .

Nous remarquons que $LT_{\geq_w}(f)$ peut très bien être un polynôme avec plus d'un terme. Par exemple, si nous prenons $w = [1, 2, 3]$ et $\mathbb{Q}_2[x, y, z]$ (avec la valuation 2-adique), alors

$$LT_{\geq_w}(x^4 + x^2y + 2y^4 + 2^{-8}z^4) = x^4 + x^2y + 2^{-8}z^4.$$

$Trop(I)$ a alors un lien particulier avec $LT_{\geq_w}(I)$:

Théorème 9.1.5 (Th. Fondamental de la Géométrie Tropicale). *Si \mathcal{K} est algébriquement clos, avec une valuation non-triviale, $Trop(I)$ est l'adhérence dans \mathbb{R}^n des points $w \in \Gamma^n$ tels que $LT_{\geq_w}(I)$ ne contient pas de monôme.*

Démonstration. Voir le Théorème 3.2.5 du livre de Maclagan et Sturmfels [MS15]. \square

Afin de calculer $LT_{\geq_w}(I)$, et pour se rapprocher d'algorithmes classiques sur les bases de Gröbner, nous utilisons un ordre monomial (classique) pour départager les égalités dans le cas où $LT_{\geq_w}(f)$ contient plus d'un monôme.

Définition 9.1.6. Soit \geq_1 un ordre monomial sur \mathcal{A} et $w \in \Gamma^n$. Nous définissons un ordre sur les termes de \mathcal{A} de la façon suivante. Soit $a, b \in \mathcal{K}$ et x^α, x^β deux monômes de \mathcal{A} , nous notons $ax^\alpha \geq bx^\beta$ si $val(a) + w \cdot \alpha < val(b) + w \cdot \beta$, ou $val(a) + w \cdot \alpha = val(b) + w \cdot \beta$ et $x^\alpha \geq_1 x^\beta$. Maintenant, si $f \in \mathcal{A}$ et I un idéal de \mathcal{A} , nous en déduisons naturellement une définition de $LT(f)$ et de $LT(I)$. Nous pouvons remarquer que $LT(I) = LT_{\geq_1}(LT_w(I))$. Nous définissons $LM(f)$ comme le monôme du terme $LT(f)$ et $LM(I)$ de la même manière. Si $G = (g_1, \dots, g_s) \in \mathcal{A}^s$ est tel que ses monômes de tête $(LM(g_1), \dots, LM(g_s))$ engendrent $LM(I)$, nous disons que G est **base de Gröbner tropicale** de I .

Nous pouvons finalement remarquer que pour calculer une partie génératrice de $LM_{\geq_w}(I)$, il est suffisant de calculer une base de Gröbner tropicale de I .

Comparaison avec les notations dans d'autres travaux Dans [CM13], le corps \mathcal{K} est tel qu'il existe un morphisme de groupes $\phi : \Gamma \rightarrow \mathcal{K}$ tel que pour tout $w \in \Gamma$, $val(\phi(w)) = w$. Si $x \in \mathcal{O}_{\mathcal{K}}$, nous notons \bar{x} sa réduction modulo $m_{\mathcal{K}}$. Nous définissons $\rho : \mathcal{K}^* \rightarrow k_{\mathcal{K}}$ par la formule $\rho(x) = x\phi(-val(x))$. ρ s'étend naturellement à $\mathcal{A} \setminus \{0\}$ par $\rho(\sum_u a_u x^u) = \sum_u \rho(a_u) x^u$. \geq_1 s'étend aussi naturellement à $\mathcal{C} = k_{\mathcal{K}}[X_1, \dots, X_n]$. Soit $w \in \Gamma^n$, alors dans [CM13], les auteurs définissent, pour tout $f \in \mathcal{A}$, $in_w = \rho(LT_{\geq_w}(f))$ et $lm(f) = LM_{\geq_1}(in_w)$. Soit $G = (g_1, \dots, g_s) \in \mathcal{A}^s$. Alors G est une base de Gröbner tropicale de $I = \langle G \rangle$ pour l'ordre sur les termes \leq si et seulement si $(in_w(g_1), \dots, in_w(g_s))$ est une base de Gröbner de $in_w(I)$ pour \leq_1 . En conséquence, calculer $LM(I)$ ou $in(I)$ fournit les même monômes. Néanmoins, nous préférons pour la suite travailler avec LM puisque nos motivations sont les calculs sur des CDVF (et non sur leurs corps résiduels).

Calculs en géométrie algébrique tropicale

Travaux disponibles sur les bases de Gröbner tropicales : Nous renvoyons au livre de Maclagan et Sturmfels [MS15] pour une introduction à la géométrie algébrique tropicale effective.

9. Une approche tropicale

Le calcul de variétés tropicales sur \mathbb{Q} avec valuation triviale est disponible à travers le package Gfan écrit par Anders Jensen (voir [Jen]), en utilisant des calculs de bases de Gröbner classiques. Cependant, pour le calcul de variétés tropicales sur des corps plus généraux, notamment avec une valuation non triviale, de telles techniques ne sont pas disponibles. C'est pourquoi Chan et MacLagan ont développé dans [CM13] une manière d'étendre la théorie des bases de Gröbner prenant en compte la valuation et qui puissent être calculées. Cette théorie des bases de Gröbner tropicales est effective et surtout permet, avec un algorithme de division adapté, un algorithme de Buchberger.

L'algorithme de Chan et MacLagan Dans leur article [CM13], Chan et MacLagan prouvent que si l'on modifie l'algorithme classique de division d'un polynôme par une famille de polynômes avec une variante de l'algorithme du cône tangent de Mora, alors on peut obtenir un algorithme de division adapté au calcul de bases de Gröbner tropicales. En effet, ils ont prouvé qu'un algorithme de Buchberger utilisant cet algorithme de division permet de calculer des bases de Gröbner tropicales.

Les idées principales de cet algorithme de division est de permettre la division par des restes partiel des divisions partielles déjà effectuées, et d'utiliser une fonction *écart* pour choisir le polynôme par lequel diviser.

Problèmes de précision L'algorithme de Chan et Macalagan s'applique, au moins de manière théorique, sur des CDVF. Par contre, il n'est pas possible de l'utiliser de manière générale sur des CDVF à précision finie. En effet, cet algorithme, de même que tous les algorithmes construits à partir de celui de Buchberger, repose sur des tests à zéro : le critère d'arrêt est celui de Buchberger. Cela n'est pas compatible avec des calculs à précision finie. Par exemple, soit F la famille de polynômes $(x^2 + xy + y^2 + (1 + O(p^N))t^2, x^2 + 2xy + 4y^2 + (1 + O(p^N))t^2, t^4) \in \mathbb{Q}_p[x, y, t]^3$, pour un certain $N \in \mathbb{N}$. Alors, l'application de l'algorithme de Chan et Macalagan (*e.g.* avec $w = (0, 0, 0)$ et l'ordre grevlex défini par $x > y > t$) conduit à des S -polynômes se réduisant en des quantités de la forme $O(\pi^{N'})xyt^2$, *i.e.* tels qu'il n'est pas possible de décider si le polynôme est le polynôme nul ou non. Ce problème de précision a lieu même avec l'usage du critère de Buchberger, et ainsi, il exclut l'utilisation d'algorithmes comme celui de Buchberger pour des calculs de bases de Gröbner sur des CDVF à précision finie.

Dans la Section suivante, nous montrons que pour le calcul de bases de Gröbner tropicales d'idéaux donnés par des polynômes homogènes, des algorithmes matriciels sont possibles.

9.2. Algorithmes F5-Matriciels tropicaux

Cette Section est consacrée à l'étude d'algorithmes F5-Matriciels pour le calcul de bases de Gröbner tropicales. Nous suivrons la méthode développée aux Chapitres 6 et 7, en présentant d'abord des algorithmes matriciels, puis un critère F5 et un premier algorithme F5-Matriciel tropical. Nous montrerons ensuite la stabilité du calcul de bases de Gröbner tropicales pour des CDVF à précision finie. Enfin, nous présenterons un algorithme F5-Matriciel avec signature pour le calcul de bases de Gröbner tropicales en précision infinie.

9.2.1. Un premier algorithme F5-Matriciel tropical

Dans cette Sous-Section, nous travaillons à précision infinie et nous cherchons à adapter au contexte tropical les algorithmes de Lazard et F5-Matriciel.

Algorithmes matriciels pour des bases de Gröbner tropicales

Suivant le formalisme développé dans la sous-section 6.2.1, nous montrons que des algorithmes matriciels sont possibles pour calculer des bases de Gröbner. En effet la remarque suivante de Lazard est toujours valide : $I \cap \mathcal{A}_d = \langle x^\alpha f_i, |\alpha| + |f_i| = d \rangle$. En particulier, il est toujours naturel d'étudier les matrices de Macaulay.

Nous rappelons que dans les algorithmes matriciels classiques pour calculer des bases de Gröbner, l'idée principale est de calculer des formes échelonnées (en lignes) des matrices de Macaulay $Mac_d(f_1, \dots, f_s)$ jusqu'à un D fixé au départ. Si D est assez grand, les lignes non nulles de la forme

échelonnée forment une base de Gröbner de I , et le premier coefficient non nul sur chaque ligne est le coefficient de tête du polynôme correspondant à cette ligne. Cela étant dit, il n'est en général pas facile de déterminer jusqu'à quel D échelonner les matrices de Macaulay, dans le cas classique comme dans le cas tropical. C'est pourquoi nous définissons les D -bases de Gröbner tropicales :

Définition 9.2.1. Soit I un idéal de \mathcal{A} . (g_1, \dots, g_l) est une D -base de Gröbner tropicale de I pour \geq si pour tout $f \in I$, homogène de degré au plus D , il existe $1 \leq i \leq l$ tel que $LT(g_i)$ divise $LT(f)$.

Calcul de formes échelonnées tropicales

Cette Sous-Sous-Section est consacrée à présenter un algorithme pour calculer $LM(\{f_1, \dots, f_i\}) \cap \mathcal{A}_d$ en calculant une forme échelonnée de $Mac_d(f_1, \dots, f_i)$. Nous pourrions lire les termes de tête pour chaque ligne de cette forme échelonnée, et pour cela, nous associons à chaque colonne une étiquette qui est le monôme correspondant à cette colonne. Ceci permettra de permuter des colonnes tout en sachant toujours à quels monômes elles correspondent et *in fine* obtenir les termes de tête des lignes.

Définition 9.2.2. Nous définissons une matrice de Macaulay étiquetée de degré d sur \mathcal{A} comme un couple (M, mon) où $M \in \mathcal{K}^{r \times \binom{n+d-1}{n-1}}$ est une matrice, et mon est une liste des $\binom{n+d-1}{n-1}$ monômes de degré d de \mathcal{A} . Chaque colonne correspond au monôme de l'étiquette de même indice.

L'algorithme 9.2.3 sur les matrices de Macaulay étiquetées permet alors de calculer, par pivot, les termes de tête pour \geq en degré d de l'idéal engendré par les lignes :

Algorithme 9.2.3 : L'algorithme d'échelonnement tropical

entrée : M , une matrice de Macaulay de degré d sur $\mathcal{A} = \mathcal{K}[X_1, \dots, X_n]$ ayant n_{lignes} lignes et n_{col} colonnes.

sortie : \widetilde{M} , la forme échelonnée tropicale de M

début

$\widetilde{M} \leftarrow M$;

si $n_{col} = 1$ ou $n_{lignes} = 0$ ou M n'a pas d'entrée non nulle **alors**

Retourner \widetilde{M} ;

sinon

Trouver i, j tels que $\widetilde{M}_{i,j}$ soit le plus grand terme $\widetilde{M}_{i,j}x^{mon_j}$ (avec plus petit i en cas d'égalité) ;

Permuter les colonnes 1 et j de \widetilde{M} , et les entrées 1 et j de mon ;

Permuter les lignes 1 et i de \widetilde{M} ;

Par pivot avec la première ligne, éliminer les coefficients sur la première colonne des autres lignes ;

Procéder récursivement sur la sous-matrice $\widetilde{M}_{i \geq 2, j \geq 2}$;

Retourner \widetilde{M} ;

Définition 9.2.4. Nous définissons la forme échelonnée (en lignes) tropicale d'une matrice de Macaulay étiquetée M comme le résultat du précédent algorithme, et nous la notons \widetilde{M} . \widetilde{M} est bien échelonnée en lignes.

Correction : $Mac_d(\widetilde{f_1}, \dots, \widetilde{f_i})$ fournit exactement les termes de tête de $\{f_1, \dots, f_i\} \cap \mathcal{A}_d$:

Proposition 9.2.5. Soit $F = (f_1, \dots, f_s)$ des polynômes homogènes de \mathcal{A} . Soit $d \in \mathbb{Z}_{>0}$ et $M = Mac_d(f_1, \dots, f_s)$, étiqueté par les monômes correspondant à ses colonnes. Soit $I = \{F\}$ l'idéal engendré par les f_i .

Soit \widetilde{M} la forme tropicale échelonnée de M . Alors les lignes de \widetilde{M} forment une base de $I \cap \mathcal{A}_d$ telle que leurs termes de tête correspondent à $LT(I) \cap \mathcal{A}_d$.

Le fait que les lignes de \widetilde{M} forment une base de $I \cap \mathcal{A}_d$ est clair : elles forment une base échelonnée (pour la base des monômes donnés dans l'ordre de l'étiquette de \widetilde{M}). Maintenant, pour ce qui est des termes de tête de $I \cap \mathcal{A}_d$, le résultat est une conséquence directe du lemme suivant :

Lemme 9.2.6. Si $ax^\alpha > b_1x^\beta$ et $ax^\alpha > b_2x^\beta$, alors $ax^\alpha > (b_1 + b_2)x^\beta$.

Conséquence : Pour calculer les polynômes d'une D -base de Gröbner tropicale de $\langle f_1, \dots, f_s \rangle$, il suffit comme pour l'algorithme de Lazard, de calculer les formes échelonnées tropicales des matrices de Macaulay (étiquetées) $Mac_d(f_1, \dots, f_s)$ pour d de 1 à D . Néanmoins, ces matrices ont un nombre de lignes et de colonnes qui croît rapidement avec d et n , et nous allons voir que le critère F5 pour réduire les tailles des matrices vaut encore pour calculer des bases de Gröbner tropicales.

Le critère F5

Nous montrons ici que, comme dans le cas classique, le critère F5 de Faugère est utilisable, et permet de supprimer la plupart des lignes des matrices de Macaulay $Mac_d(f_1, \dots, f_s)$ n'apportant pas d'information utile pour le calcul de $LT(I)$.

Pour tout $j \in \llbracket 1, s \rrbracket$, nous notons I_j l'idéal (f_1, \dots, f_j) . Alors, Faugère a montré dans [Fau02] que pour un ordre monomial classique, si l'on sait quels monômes x^α sont dans $LM(I_{i-1})$, alors on peut supprimer les lignes correspondant aux $x^\alpha f_i$ des matrices de Macaulay $Mac(f_1, \dots, f_i)$ et toujours pouvoir calculer $LM(I_i)$ (voir aussi la sous-section 6.2.1). Nous montrons ici que ce critère est directement compatible avec notre définition de LM :

Théorème 9.2.7 (Critère F5). Pour tout $i \in \llbracket 1, s \rrbracket$,

$$I_i \cap \mathcal{A}_d = Vect(\{x^\alpha f_k, t.q. 1 \leq k \leq i, |x^\alpha f_k| = d \text{ et } x^\alpha \notin LM(I_{k-1})\}).$$

Démonstration. Nous utilisons le fait suivant, qui peut se prouver par récurrence. Soit (f_1, \dots, f_i) des polynômes homogènes de \mathcal{A} de degré d_1, \dots, d_i . Soit $a_{\alpha_1}x^{\alpha_1}, \dots, a_{\alpha_u}x^{\alpha_u}$ les termes de têtes des lignes de $Mac_{d-d_i}(f_1, \dots, f_{i-1})$, triés par ordre décroissant (selon \leq). Soit x^{β_j} les autres monômes de degré $d - d_i$ (i.e. les monômes qui ne sont pas un monôme de tête de $(f_1, \dots, f_{i-1}) \cap \mathcal{A}_{d-d_i}$). Alors, pour tout k , la ligne $x^{\alpha_k} f_i$ de $Mac_d(f_1, \dots, f_i)$ est une combinaison linéaire de lignes de la forme $x^{\alpha_k + \gamma} f_i$ ($k' > 0$), $x^{\beta_j} f_i$ et $x^\gamma f_j$ ($j < i$) de $Mac_d(f_1, \dots, f_i)$. \square

Ainsi, nous voyons clairement avec le critère F5 quelles lignes supprimer des matrices de Macaulay. La prochaine sous-section explique comment utiliser effectivement ce critère.

Un premier algorithme F5-Matriciel tropical

Un algorithme F5-Matriciel Nous appliquons le cadre de la Sous-Section 6.2.1 avec l'algorithme d'échelonnement tropical 9.2.3 et nous obtenons l'algorithme suivant :

Algorithme 9.2.8 : Un premier algorithme F5-Matriciel

entrée : $F = (f_1, \dots, f_s) \in \mathcal{A}^s$, homogènes de degrés $d_1 \leq \dots \leq d_s$, et $D \in \mathbb{N}$

sortie : $(g_1, \dots, g_k) \in \mathcal{A}^k$, une D -base de Gröbner tropicale de $\{F\}$.

début

$G \leftarrow F$;

pour $d \in \llbracket 0, D \rrbracket$ **faire**

$\widetilde{\mathcal{M}}_{d,0} := \emptyset$;

pour $i \in \llbracket 1, s \rrbracket$ **faire**

$\mathcal{M}_{d,i} := \widetilde{\mathcal{M}}_{d,i-1}$;

pour α tel que $|\alpha| + d_i = d$ **faire**

si x^α n'est pas le terme de tête d'une ligne de $\widetilde{\mathcal{M}}_{d-d_i,i-1}$ **alors**

Ajouter $x^\alpha f_i$ à $\mathcal{M}_{d,i}$;

Calculer $\widetilde{\mathcal{M}}_{d,i}$, la forme échelonnée tropicale de $\mathcal{M}_{d,i}$ par l'Algorithme 9.2.3;

Ajouter à G toutes les lignes avec un nouveau terme de tête. ;

Retourner G ;

Correction Il suffit de prouver que pour tout $d \in \llbracket 0, D \rrbracket$ et $i \in \llbracket 1, s \rrbracket$, $\text{Im}(\mathcal{M}_{d,i}) = I_i \cap \mathcal{A}_d$. Nous pouvons le faire par récurrence sur d et i . Nous remarquons qu'il n'y a rien à prouver pour $i = 1$ et d quelconque. Maintenant, supposons qu'il existe un $i \in \llbracket 1, s \rrbracket$ tel que pour tout j tel que $1 \leq j < i$ et pour tout d , on a $0 \leq d \leq D$, $\text{Im}(\mathcal{M}_{d,j}) = I_j \cap \mathcal{A}_d$. Alors, pour un tel i , le premier entier d tel que $\mathcal{M}_{d,i} \neq \mathcal{M}_{d,i-1}$ est d_i . Soit d tel que $d_i \leq d \leq D$. Alors, avec le Théorème 9.2.7 :

$$I_i \cap \mathcal{A}_d = \text{Im}(\mathcal{M}_{d,i-1}) + \text{Vect}(\{x^\alpha f_i, \text{ t.q. } x^\alpha \notin LM(I_{i-1})\}). \quad (9.1)$$

De plus, par l'hypothèse de récurrence et la preuve de la correction de l'algorithme d'échelonnement (Proposition 9.2.5), les termes de tête de $I_{i-1} \cap \mathcal{A}_{d-d_i}$ sont exactement les termes de têtes des lignes de $\mathcal{M}_{d-d_i,i-1}$. Ainsi, les lignes que l'on ajoute à $\mathcal{M}_{d,i-1}$ pour construire $\mathcal{M}_{d,i}$ sont exactement les $x^\alpha f_i$ tels que $x^\alpha \notin LM(I_{i-1})$. Enfin, nous remarquons que $\text{Im}(\mathcal{M}_{d,i}) = \text{Im}(\mathcal{M}_{d,i-1})$. En conséquence, $\text{Im}(\mathcal{M}_{d,i})$ contient les deux termes de la somme vectorielle de (9.1), et comme il est clairement inclus dans $I_i \cap \mathcal{A}_d$, nous avons prouvé que $I_i \cap \mathcal{A}_d = \text{Im}(\mathcal{M}_{d,i})$. Pour conclure, nous observons que, du fait de la correction de l'algorithme de calcul de forme tropicale échelonnée (Proposition 9.2.5), les termes de tête des lignes de $\mathcal{M}_{d,i}$ correspondent bien aux termes de têtes des polynômes de $I_i \cap \mathcal{A}_d$.

Suites régulières et complexité

Syzygies principales et régularité Le comportement de l'algorithme F5-Matriciel tropical vis-à-vis des syzygies principales est le même que celui de l'algorithme F5-Matriciel classique. Il est résumé dans la proposition suivante et dans son corollaire.

Proposition 9.2.9. *Si une ligne se réduit à zéro durant l'exécution de l'algorithme F5-Matriciel tropical, alors la syzygie qu'elle engendre n'est pas dans le module des syzygies principales.*

Démonstration. Soit $\sum_{j=1}^i a_j f_j$ avec $a_j \in \mathcal{A}$ une syzygie de (f_1, \dots, f_i) . Si $a_j \neq 0$ et si cette syzygie est principale, alors $a_i \in I_{i-1}$ et $LM(a_i) \in LM(I_{i-1})$. Grâce au critère F5, il n'y a pas de ligne de la forme $x^\alpha f_i$ avec $x^\alpha \in LM(I_{i-1})$ dans la matrice de Macaulay $\mathcal{M}_{d,i}$. Ainsi, une telle syzygie ne peut être produite durant la réduction de $\mathcal{M}_{d,i}$. \square

Corollaire 9.2.10. *Si la suite (f_1, \dots, f_s) est régulière, alors aucune ligne des matrices de Macaulay réduites durant l'exécution de l'algorithme F5-Matriciel tropical n'est réduite à zéro. En d'autres termes, les matrices $\mathcal{M}_{d,i}$ sont injectives (à gauche), et ont au plus autant de lignes que de colonnes.*

Démonstration. Pour une suite régulière de polynômes homogènes, toutes les syzygies sont principales. Nous renvoyons pour plus de détails à [EM07] page 69. \square

Complexité La complexité du calcul d'une forme échelonnée tropicale de rang r avec n_{lignes} lignes et n_{col} colonnes est en $O(r \times n_{\text{rows}} \times n_{\text{cols}})$ opérations dans \mathcal{K} . Nous en déduisons les complexités suivantes pour l'algorithme 9.2.8 :

- $O\left(s^2 D \binom{n+D-1}{D}^3\right)$ opérations dans \mathcal{K} , pour $D \rightarrow +\infty$.
- $O\left(s D \binom{n+D-1}{D}^3\right)$ opérations dans \mathcal{K} , pour $D \rightarrow +\infty$, dans le cas particulier où (f_1, \dots, f_s) régulière, du fait du corollaire 9.2.10.

La majoration que nous obtenons est la même que celle que nous obtenons en Sous-Sous-Section 6.2.1 pour l'algorithme F5-Matriciel sans étiquette (6.2.12). Là encore, la complexité subit essentiellement l'ajout d'un facteur s par rapport à la variante de l'algorithme F5-Matriciel avec signature (6.2.22). Ceci vient du fait que nous calculons entièrement la forme échelonnée tropicale pour chaque nouveau $\mathcal{M}_{d,i}$. En d'autres termes, nous ne tenons pas compte du fait que lors de la construction de la matrice $\mathcal{M}_{d,i}$, $\mathcal{M}_{d,i-1}$ est déjà sous forme échelonnée. Ce choix est motivé par la précision : il permet de ne pas accumuler des erreurs dues à des pivots de trop grande valuation quand on passe de i à $i+1$. Nous verrons en Sous-Section 9.2.4 qu'il est possible, lorsqu'on travaille en précision infinie, de privilégier la vitesse en utilisant effectivement le fait que $\mathcal{M}_{d,i-1}$ est déjà sous forme échelonnée.

9. Une approche tropicale

Bornes sur D Pour ce qui est d'avoir une majoration sur le degré D à partir duquel les D -bases de Gröbner sont des bases de Gröbner, Chan a prouvé dans [Cha13] (Theorem 3.3.1) que $D = 2(d^2/2 + d)^{2^{n-2}}$, avec $d = \max_i d_i$, est suffisant. Nous remarquons aussi que si (f_1, \dots, f_n) est une suite régulière, alors tous les monômes de degré plus grand que la borne de Macaulay $\sum_i (d_i - 1) + 1$ sont dans $LM(I)$. C'est une conséquence directe du fait que l'on connaît la fonction de Hilbert d'une suite régulière. Ainsi, nous pouvons énoncer la proposition suivante :

Proposition 9.2.11. *Si $(f_1, \dots, f_n) \in \mathcal{A}^n$ est une suite régulière de polynômes homogènes, toutes les D -bases de Gröbner sont des bases de Gröbner dès que $D \geq \sum_i (|f_i| - 1) + 1$.*

Un exemple Nous reprenons l'Exemple 6.2.15, mais cette fois-ci, dans un contexte tropical :

Exemple 9.2.12. Nous appliquons l'Algorithme 9.2.8 sur $F = (f_1, f_2, f_3) \in \mathbb{Q}[x, y, z]$ avec valuation 2-adique, poids $w = [0, 0, 0]$ et l'ordre grevlex pour briser les égalités, et $f_1 = 2x + z$, $f_2 = x^2 + y^2 - 2z^2$ et $f_3 = 4y^2 + yz + 8z^2$. Nous prenons $D = 3$, la borne de Macaulay. Nous partons de $G = (f_1, f_2, f_3)$.

$$\begin{array}{c}
 \begin{array}{c} z \ y \ x \\ f_1 \mid 1 \ 0 \ 2 \end{array} \\
 \text{En degré 1, nous obtenons } \widetilde{\mathcal{M}}_{1,1} = \widetilde{\mathcal{M}}_{1,2} = \widetilde{\mathcal{M}}_{1,3} \text{ qui sont}
 \end{array}$$

$$\begin{array}{c}
 \begin{array}{c} xz \ yz \ z^2 \ x^2 \ xy \ y^2 \\ zf_1 \mid \quad \quad 1 \ -4 \\ yf_1 \mid \quad 1 \quad \quad \quad 2 \\ xf_1 \mid 1 \quad \quad \quad 2 \end{array} \\
 \text{En degré 2, } \widetilde{\mathcal{M}}_{2,1} \text{ est}
 \end{array}$$

$$\begin{array}{c}
 \begin{array}{c} x^2 \ xz \ yz \ z^2 \ xy \ y^2 \\ zf_1 \mid \quad \quad \quad -7 \quad 4 \\ yf_1 \mid \quad \quad 1 \quad \quad 2 \\ xf_1 \mid \quad 1 \quad \quad 4 \quad -2 \\ f_2 \mid 1 \quad \quad -2 \quad 1 \\ f_3 \mid \quad \quad \quad -2 \ 60/7 \end{array} \\
 \text{De même : } \widetilde{\mathcal{M}}_{2,3} \text{ est}
 \end{array}$$

$$\begin{array}{c}
 \begin{array}{c} x^2z \ xyz \ y^2z \ xz^2 \ yz^2 \ z^3 \ xy^2 \ y^3 \ z^3 \ x^2y \\ z^2f_1 \mid \quad \quad \quad \quad \quad 1 \quad \quad \quad 8 \\ y^2f_1 \mid \quad \quad \quad \quad \quad 1 \quad \quad \quad -4 \\ x^2f_1 \mid \quad \quad \quad 1 \quad \quad \quad \quad \quad -4 \\ y^2f_1 \mid \quad \quad \quad 1 \quad \quad \quad \quad \quad 2 \\ xyf_1 \mid \quad 1 \quad \quad \quad \quad \quad 1 \quad \quad \quad 2 \\ x^2f_1 \mid 1 \quad \quad \quad \quad \quad 1 \quad \quad \quad 2 \end{array} \\
 \text{la matrice suivante :}
 \end{array}$$

$$\begin{array}{c}
 \text{Grâce au critère F5, nous pouvons écarter } zf_2 \text{ et seulement ajouter } yf_2 \text{ et } xf_2 \text{ pour définir } \mathcal{M}_{3,2}.
 \end{array}$$

$$\begin{array}{c}
 x^3 \quad x^2y \quad x^2z \quad xyz \quad y^2z \quad xz^2 \quad yz^2 \quad z^3 \quad xy^2 \quad y^3 \\
 \begin{array}{l}
 z^2f_1 \\
 y^zf_1 \\
 x^zf_1 \\
 y^2f_1 \\
 xyf_1 \\
 x^2f_1 \\
 zf_2 \\
 yf_2
 \end{array}
 \left| \begin{array}{cccccccccc}
 & & & & & & & 1 & 8/7 & \\
 & & & & & & -7 & & & 4 \\
 & & & & & -7 & & & 4 & \\
 & & & & 1 & & & & 2 & \\
 & & & 1 & & & 4 & & & -2 \\
 & & 1 & & & 4 & & & -2 & \\
 & 1 & & & & & -2 & & & 1 \\
 1 & & & & & -2 & & & 1 &
 \end{array} \right.
 \end{array}$$

Nous obtenons ainsi $\widetilde{\mathcal{M}}_{3,2}$ comme ceci :

Enfin, à nouveau grâce au critère F5, nous pouvons écarter zf_3 et nous obtenons $\widetilde{\mathcal{M}}_{3,3}$ comme :

$$\begin{array}{c}
 x^3 \quad x^2y \quad x^2z \quad xyz \quad y^2z \quad xz^2 \quad yz^2 \quad z^3 \quad xy^2 \quad y^3 \\
 \begin{array}{l}
 z^2f_1 \\
 y^zf_1 \\
 x^zf_1 \\
 y^2f_1 \\
 xyf_1 \\
 x^2f_1 \\
 zf_2 \\
 yf_2 \\
 zf_3 \\
 yf_3
 \end{array}
 \left| \begin{array}{cccccccccc}
 & & & & & & & 1 & 8/7 & \\
 & & & & & & -7 & & & 4 \\
 & & & & & -7 & & & 4 & \\
 & & & & 1 & & & & 2 & \\
 & & & 1 & & & 4 & & & -2 \\
 & & 1 & & & 4 & & & -2 & \\
 & 1 & & & & & -2 & & & 1 \\
 1 & & & & & -2 & & & 1 & \\
 & & & & & & -2 & 60/7 & & \\
 & & & & & & & 1786/49 & &
 \end{array} \right|
 \end{array}$$

9.2.2. Le cas des CDVF à précision finie

Problèmes de précision

Nous rappelons que dans un CDVF à précision finie K , nous nous intéressons à des calculs sur des approximations x d'éléments de K qui s'écrivent sous la forme $x = \sum_{k \geq l}^{m-1} a_k \pi^k + O(\pi^m)$. m est appelé la précision sur x .

Nous remarquons que si la précision sur les coefficients de $f \in A$ n'est pas assez grande, il peut arriver qu'on ne puisse pas déterminer quel est le terme de tête de f . Par exemple, sur $\mathbb{Q}_p[X_1, X_2]$, avec $w = (0, 4)$ et l'ordre lexicographique, on ne peut pas comparer $O(p^2)X_1$ et X_2 . Cependant, à part si le premier coefficient est nul, lorsque la précision sur les coefficients est suffisamment élevée, un tel problème n'apparaît pas. Dans ce qui suit, nous explicitons une précision suffisante sur les coefficients d'un polynôme donné pour que l'on puisse déterminer quel est son terme de tête.

Proposition 9.2.13. *Soit $f \in A$ un polynôme homogène et soit aX^α son terme de tête. Alors une précision $\text{val}(a) + \max_{|\beta|=d} ((\alpha - \beta) \cdot w)$ sur les coefficients de f est suffisante pour montrer que aX^α est bien le terme de tête de f .*

Démonstration. Il suffit de remarquer que $O(p^n)X^\beta < aX^\alpha$ si et seulement si $n > \text{val}(a) + (\alpha - \beta) \cdot w$. \square

Calcul de formes échelonnées

Suites régulières : Comme souvent lorsqu'on considère des calculs avec des coefficients connus avec une précision finie, on ne peut pas décider si un nombre sans chiffre de précision, comme $O(\pi^k)$, est zéro ou non. Heureusement, grâce au corollaire 9.2.10, lorsque les polynômes en entrée forment une suite régulière, les matrices mises en jeu dans l'algorithme F5-Matriciel tropical sont

9. Une approche tropicale

injectives. Cela signifie que si la précision est suffisante, il n'y aura pas de problème pour trouver les pivots (bien non nuls) lors de la mise sous forme échelonnée tropicale, et nous pourrons bien déterminer quels sont les termes de tête sur chaque ligne.

Nous pouvons donner une estimation d'un majorant de la précision nécessaire pour pouvoir calculer une D -base de Gröbner à partir d'une telle suite :

Une précision suffisante :

Proposition 9.2.14. *Soit M une matrice de Macaulay étiquetée avec coefficients dans R et de degré d . Soit a_1, \dots, a_u les pivots choisis durant le calcul de la forme échelonnée tropicale. Soit x^{α_k} les monômes correspondant. Soit $prec$ défini par :*

$$prec = \sum_k val(a_k) + \max_k val(a_k) + \max_{k, |\beta|=d} (\alpha_k - \beta) \cdot w.$$

Alors, si les coefficients de la matrice sont connus avec une même précision $O(\pi^{prec})$, la forme échelonnée tropicale de M est calculée sans problème de précision, et la perte de précision est majorée par $\sum_k val(a_k)$.

Démonstration. Nous commençons par considérer une matrice de Macaulay étiquetée M avec coefficients dans R tous connus à la précision $O(\pi^l)$, et nous supposons dans un premier temps qu'il n'y a pas de problème de précision lors de l'échelonnement de la matrice. Ainsi, nous analyserons d'abord quelle est la perte de précision lorsqu'on pivote et nous pourrons ensuite donner une estimation d'une précision suffisante. Ainsi, pour pivoter, nous souhaitons obtenir un vrai zéro pour le coefficient $M_{i,j} = \varepsilon \pi^{n_1} + O(\pi^n)$, en pivotant avec le pivot $piv = \mu \pi^{n_0} + O(\pi^n)$ qui est sur la ligne L , avec $n_0, n_1 < n$ des entiers, et $\varepsilon = \sum_{k=0}^{n-n_1-1} a_k \pi^k$, $\mu = \sum_{k=0}^{n-n_0-1} b_k \pi^k$, avec $a_k, b_k \in S_K$, et $a_0, b_0 \neq 0$. Nous pouvons remarquer que de par la manière dont sont choisis les pivots, nécessairement, $n_0 \leq n_1$. Maintenant, nous obtenons ce zéro par l'opération suivante sur la i -ème ligne L_i :

$$L_i \leftarrow L_i - \frac{M_{i,j}}{piv} L = L_i + (\varepsilon \mu^{-1} \pi^{n_1-n_0} + O(\pi^{n-n_0})) L,$$

en même temps que l'opération formelle $M_{i,j} \leftarrow 0$. En effet, $\frac{M_{i,j}}{piv} = \frac{\varepsilon \pi^{n_1} + O(\pi^n)}{\mu \pi^{n_0} + O(\pi^n)}$, donc $\frac{M_{i,j}}{piv} = \varepsilon \mu^{-1} \pi^{n_1-n_0} + O(\pi^{n-n_0})$. En conséquence, une fois que le premier pivot est choisi et que les autres coefficients sur la première colonne ont été réduits à zéro, les coefficients de la sous-matrice $\widetilde{M}_{i \geq 2, j \geq 2}$ sont connus à précision $O(\pi^{l-val(a_1)})$. Nous pouvons alors procéder récursivement et prouver qu'une fois finie la mise sous forme échelonnée tropicale, les coefficients de \widetilde{M} sont connus avec précision $O(\pi^{l-val(a_1 \times \dots \times a_u)})$. Pour pouvoir déterminer quels sont les pivots lors du calcul de la forme échelonnée, la Proposition 9.2.13 montre qu'il suffit que $l - val(a_1 \times \dots \times a_u)$ soit plus grand que $\max_{k, |\beta|=d} (\alpha_k - \beta) \cdot w$. Ainsi, le résultat est prouvé. \square

Algorithme F5-Matriciel tropical et précision finie

Nous appliquons cette étude du calcul de la forme échelonnée tropicale dans le cas de la précision finie pour prouver la Proposition 9.1.2 concernant l'algorithme F5-Matriciel sur des CDVF à précision finie. Pour faciliter cette étude, et seulement dans cette sous-section, l'étape $\mathcal{M}_{d,i} := \widetilde{\mathcal{M}_{d,i-1}}$ dans l'algorithme 9.2.8 est remplacée par $\mathcal{M}_{d,i} := \mathcal{M}_{d,i-1}$. Cela ne pose aucun problème par rapport au calcul de bases de Gröbner car ces deux matrices ont les mêmes dimensions et la même image. Nous définissons d'abord des bornes sur une précision initiale suffisante et la perte de précision lors du calcul. Soit $(f_1, \dots, f_s) \in B^s$ une suite régulière de polynômes homogènes.

Définition 9.2.15. Soit $d \geq 1$ et $1 \leq i \leq s$. Soit $x^{\alpha_1}, \dots, x^{\alpha_u}$ les monômes des termes de tête de $\langle f_1, \dots, f_i \rangle \cap A_d$. Soit $\Delta_{d,i}$ le mineur de $\mathcal{M}_{d,i}$ sur les colonnes correspondant aux x^{α_l} qui, parmi ces mineurs, atteint la plus petite valuation. Soit

$$\square_{d,i} = 2\Delta_{d,i} + \max_{k, |\beta|=d} (\alpha_k - \beta) \cdot w.$$

Nous définissons

$$prec_{MF5trop}((f_1, \dots, f_s), D, \geq) = \max_{d \leq D, i} \square_{d,i},$$

et

$$loss_{MF5trop}((f_1, \dots, f_s), D, \geq) = \max_{d \leq D, i} \Delta_{d,i}.$$

Grâce à la Proposition 9.2.14, ces bornes sont suffisantes pour la Proposition 9.1.2.

En outre, nous pouvons préciser le cas particulier où $w = 0$:

Proposition 9.2.16. *Si $w = 0$, alors la perte de précision correspond au mineur maximal (en taille) de $\mathcal{M}_{d,i}$ qui atteint la plus petite valuation. En particulier, $w = 0$ correspond à la plus petite $loss_{MF5trop}$. De plus, dans ce cas-ci,*

$$prec_{MF5trop}((f_1, \dots, f_s), D, \geq) = 2loss_{MF5trop}((f_1, \dots, f_s), D, \geq).$$

Ce cas est particulièrement intéressant car les calculs sur les matrices de Macaulay effectués dans le cas $w = 0$ sont très similaires à ceux que l'on effectue pour le calcul d'une forme normale de Smith.

Précision vs Complexité

Nous pouvons remarquer que si nous voulons atteindre la plus petite perte de précision, nous pouvons décider d'abandonner la suppression de lignes par le critère F5 et utiliser l'algorithme de calcul de formes échelonnées tropicales directement sur les matrices de Macaulay complète, jusqu'à ce qu'assez de lignes indépendantes (et échelonnées) soient obtenues. Le rang de ces matrices peut être calculé grâce au critère F5 et au corollaire 9.2.7 en échelonnant les matrices itérativement en d et i . De cette manière, nous serons assurés de choisir des pivots générant la plus petite perte de précision possible (par mise sous forme échelonnée) sur les $Mac_d(f_1, \dots, f_i)$ (du fait que ces choix correspondent au mineur sur les colonnes correspondant à un monôme de tête de plus petite valuation). Cependant, un tel algorithme serait, bien sûr, plus gourmand en temps à cause de toutes les lignes se réduisant à zéro, et serait en $O\left(s^2 D \binom{n+D-1}{D}^3\right)$ même pour des suites régulières.

Comparaison avec les bases de Gröbner classiques

Nous pouvons maintenant comparer notre étude de la précision lors du calcul de bases de Gröbner tropicales avec celle réalisée pour des bases de Gröbner classiques.

Nous rappelons que, d'après le Théorème 7.1.1, pour $F \in A^s$ et sous les hypothèses de régularité **H1** et **H2**, une précision $prec_{F5M}(F, D, \geq)$, essentiellement donnée par des mineurs des matrices de Macaulay, est suffisante pour le calcul d'une base de Gröbner de $\langle F \rangle$ pour l'ordre monomial \geq , par un algorithme F5-Matriciel faible. La perte de précision lors du calcul est aussi majorée par $prec_{F5M}(F, D, \geq)$.

Nous remarquons que pour le calcul de bases de Gröbner tropicales l'hypothèse de structure **H2** est, en quelque sorte, compensée par une condition sur la précision qui peut être plus dure : $\max_k val(a_k) + \max_{k, |\beta|=d} (\alpha_k - \beta) \cdot w$, afin qu'il n'y ait pas de problème de précision dans la détermination des pivots et des termes de tête des lignes. Ceci amène à une précision suffisante $prec_{MF5trop}(F, D, \geq)$ qui peut être plus grande que $prec_{MF5}$ mais il n'y a plus de problème de position des coefficients non nuls, ni de besoin de l'hypothèse **H2**.

Ainsi, pour calculer des bases de Gröbner tropicales sur un CDVF à précision finie (avec valuation non-triviale), la seule hypothèse de structure que l'on demande est la régularité, **H1**. Cette dernière est clairement générique, alors que dans le cas classique, **H1** et **H2** ne sont au mieux que conjecturalement génériques dans des cas particuliers comme celui de l'ordre grevlex (conjecture de Moreno-Socias). En conséquence, le calcul de bases de Gröbner tropicale peut, d'un côté, demander une précision en entrée plus grande que pour des bases de Gröbner classiques, mais d'un autre côté, ce calcul est génériquement possible, alors que ce n'est pas quelque chose de connu pour les bases de Gröbner classiques.

Enfin, lorsque le poids w est zéro, grâce à la Proposition 9.2.16, nous obtenons la plus petite majoration de la perte de précision définie par des mineurs des matrices de Macaulay. Elle est en

9. Une approche tropicale

particulier inférieure aux majorations définies dans le cas classique au Chapitre 7. C'est donc ce choix de poids que nous préconisons si l'on cherche à calculer une base de Gröbner tropicale en minimisant la perte de précision.

Exemple 9.2.17. Nous pouvons reprendre l'exemple utilisé pour 6.2.15, 7.2.15 et 9.2.12 en partant de $F = (f_1, f_2, f_3) \in \mathbb{Q}_2[x, y, z]$ connus à précision $O(2^{10})$. Le comportement de l'algorithme est alors le même que pour l'exemple 9.2.12, et nous remarquons qu'aucun pivot de valuation strictement positive n'est utilisé pour éliminer un coefficient. Autrement dit, il n'y a dans les faits aucune perte de précision, contrairement aux deux chiffres perdus lors de l'exemple 7.2.15.

Une autre comparaison intéressante à faire est celle avec la méthode TSV développée par Faugère et Liang dans [FL07, FL11a, FL11b]. Dans celle-ci, les auteurs remplacent un éventuel terme de tête εx^α avec ε trop petit par εy où y est une nouvelle variable, $y < x^\alpha$ et ajoutent au système de polynômes $x^\alpha - y$. Appliquée plus concrètement à un algorithme F5-Matriciel, cette stratégie a de similaire avec les méthodes tropicales développées dans ce chapitre le fait qu'elle évite les pivots sur lesquels la précision est trop faible en ajoutant une variable et en tordant l'ordre monomial pour que les termes correspondants ne soient plus termes de tête. Par rapport à la méthode TSV, nous remarquons que le calcul de bases de Gröbner tropicales permet de s'affranchir de l'ajout de nouvelles variables, ainsi que de l'éventuel borne à fournir sur le nombre de substitutions de termes par des variables à réaliser. Ceci permet une meilleure estimation *a priori* de la complexité. Il serait intéressant de pousser plus en avant la comparaison, notamment concernant la gestion de la précision dans cette méthode TSV.

9.2.3. Implémentation

Une implémentation jouet en Sage [S⁺11] des algorithmes précédents est disponible sur http://perso.univ-rennes1.fr/tristan.vaccon/toy_F5.py. Comme le but de cette implémentation est l'étude de la précision, elle n'est pas nécessairement optimisée pour ce qui est du temps de calcul. Nous avons appliqué l'algorithme F5-Matriciel à des polynômes homogènes de degrés donnés et avec des coefficients pris aléatoirement dans \mathbb{Z}_p (pour la mesure de Haar) : f_1, \dots, f_s , de degrés d_1, \dots, d_s dans $\mathbb{Z}_p[X_1, \dots, X_s]$, connus à précision $O(p^{30})$, avec un poids donné w et l'ordre grevlex pour les cas d'égalité, et avec D la borne de Macaulay. Cette expérience est réalisé 20 fois pour chaque choix de paramètres et sont notés la perte de précision maximale, moyenne, et le nombre d'échecs (*i.e.* le calcul n'a pas pu être fini du fait d'une précision insuffisante). Ces résultats sont comparés avec ceux donnés par l'algorithme F5-Matriciel Faible 7.2.7 pour grevlex avec les même paramètres (il s'agit des lignes avec "grevlex" dans le tableau). Les résultats sont consignés dans le tableau suivant :

| $d =$ | w | D | p | perte maximale | perte moyenne | échecs |
|-----------|------------|-----|-----|----------------|---------------|--------|
| [3,4,7] | grevlex | 12 | 2 | 9 | 0.1 | 0 |
| [3,4,7] | [1,-3,2] | 12 | 2 | 11 | 0.1 | 0 |
| [3,4,7] | [0,0,0] | 12 | 2 | 0 | 0 | 0 |
| [3,4,7] | [1,-3,2] | 12 | 7 | 3 | .02 | 0 |
| [3,4,7] | [0,0,0] | 12 | 7 | 0 | 0 | 0 |
| [2,3,4,5] | grevlex | 11 | 2 | 9 | 1.6 | 2 |
| [2,3,4,5] | [1,4,1,-1] | 11 | 2 | 13 | 0.2 | 0 |
| [2,3,4,5] | [0,0,0,0] | 11 | 2 | 0 | 0 | 0 |
| [2,3,4,5] | [1,4,1,1] | 11 | 7 | 5 | 0.02 | 0 |

Nous pouvons remarquer que ces résultat suggèrent à nouveau que la perte de précision est plus faible lorsque l'on travaille avec un p plus grand. Cela reste raisonnable puisque la perte de précision vient de pivot ayant une valuation strictement positive, tandis que la probabilité que $val(x) = 0$ pour $x \in \mathbb{Z}_p$ est $\frac{p-1}{p}$. Ces résultats confirment aussi le fait que $w = [0, \dots, 0]$ produit des pertes de précision significativement plus faibles (et souvent inexistantes).

9.2.4. Un algorithme F5-Matriciel plus rapide

Dans cette Sous-Section, nous montrons qu'il est possible de réaliser dans un contexte tropical une adaptation de l'algorithme classique F5-Matriciel avec signature 6.2.22. Cette variante de l'algorithme

F5-Matriciels est caractérisé par l'usage du fait que $\widetilde{\mathcal{M}_{d,i-1}}$ est déjà sous forme échelonnée lorsque l'on construit $\mathcal{M}_{d,i}$.

À cet effet, nous introduisons les concepts d'étiquette et de signature de polynômes, ainsi que le calcul d'une forme LUP tropicale qui donnera les termes de têtes des lignes des Matrices de Macaulay sans modifier les signatures.

Nous n'étudions cet algorithme que du point de vue de la précision infinie. Il n'est en effet pas particulièrement adapté à la précision finie car, par exemple, on n'y choisit pas ses pivots.

Étiquettes et signatures

Définition 9.2.18. Soit $(f_1, \dots, f_s) \in \mathcal{A}^s$. Un *polynôme étiqueté* est un couple (u, P) avec $u = (l_1, \dots, l_s) \in \mathcal{A}^s$, $P \in \mathcal{A}$ et $\sum_{i=1}^s l_i f_i = P$. u est appelé l'*étiquette* de ce polynôme étiqueté.

Nous notons (e_1, \dots, e_s) la base canonique de \mathcal{A}^s . Si $u = (l_1, \dots, l_i, 0, \dots, 0)$ avec $l_i \neq 0$, alors la *signature* du polynôme étiqueté (u, p) , notée $\text{sign}((u, p))$, est $(HM(l_i), i)$, en suivant la définition suivante : $HM(l_i)$ est le plus grand monôme, selon \leq , qui apparaît dans l_i avec coefficient non-nul.

Remarque 9.2.19. Nous remarquons que dans la définition de signature, nous ne prenons *pas* en compte les valuations des coefficients dans l'étiquette. C'est pourquoi nous avons choisi la notation $HM(l_i)$ plutôt que $LT(l_i)$ ou $LM(l_i)$. $HM(l_i)$ n'est pas, en général, le monôme du terme de tête de l_i .

Définition 9.2.20. L'ensemble des signatures $\{\text{monômes de } R\} \times \{1, \dots, s\}$ peut être munie d'un ordre total défini de la manière suivante : $(x^\alpha, i) \leq (x^\beta, k)$ si $i < k$, ou $x^\alpha \leq x^\beta$ et $i = k$.

Les signatures sont compatibles avec les opérations usuelles sur les polynômes :

Proposition 9.2.21. Soit (u, p) un polynôme étiqueté, $(x^\alpha, i) = \text{sign}((u, l))$ et soit x^β un monôme de \mathcal{A} . Alors

$$\text{sign}((x^\beta u, x^\beta p)) = (x^\alpha x^\beta, i).$$

Si (v, q) est un autre polynôme étiqueté tel que $\text{sign}((v, q)) < \text{sign}((u, p))$, et si $\mu \in \mathcal{K}$, alors $\text{sign}((u + \mu v, p + \mu q)) = \text{sign}((u, p))$.

Calcul d'une forme LUP préservant la signature

À partir de maintenant et durant toute cette Sous-Section, nous associons aux matrices de Macaulay étiquetées les étiquettes et les signatures de chaque ligne. Nous demandons de plus que les lignes soient triées par signature croissante (la première ligne ayant la plus petite signature). Lorsque nous effectuerons des opérations sur les lignes d'une matrice de Macaulay étiquetée, les opérations seront répercutées sur les étiquettes et signatures correspondants.

L'algorithme : L'algorithme suivant calcule la partie U de ce que nous appellerons une forme LUP tropicale. Elle suffit pour déterminer l'idéal de tête des lignes tout en préservant les signatures.

Algorithme 9.2.22 : L'algorithme LUP tropical

entrée : \widetilde{M} , une matrice de Macaulay de degré d sur \mathcal{A} , ayant n_{lignes} lignes et n_{col} colonnes.

sortie : \widetilde{M} , le U d'une forme LUP tropicale de M

début

$\widetilde{M} \leftarrow M$;

si M n'a pas de coefficient non-nul **alors**

 Retourner \widetilde{M} ;

sinon

pour $i = 1$ jusqu'à n_{lignes} **faire**

 Trouver j tel que $\widetilde{M}_{i,j}$ ait le plus grand terme $\widetilde{M}_{i,j}x^{mon_j}$ de sa ligne ;

 Permuter les colonnes 1 et j de \widetilde{M} , et les entrées 1 et j de mon ;

 Par pivot avec la première ligne, éliminer les coefficients sur la première colonne des autres lignes ;

 Procéder récursivement sur la sous-matrice $\widetilde{M}_{i \geq 2, j \geq 2}$;

 Retourner \widetilde{M} ;

Nous remarquons qu'à la sortie de l'algorithme, il existe une matrice unipotente triangulaire inférieure L , une matrice de permutation P , tels que $\widetilde{M} = LMP$, \widetilde{M} est sous forme échelonnée (en lignes). De plus, comme à chaque ligne L_i , nous avons seulement ajouté une combinaison linéaire de lignes L_j , avec $j < i$, qui ont une signature strictement plus petite que L_i , les signatures sont conservées. En outre,

Proposition 9.2.23. *Pour tout $1 \leq i \leq n_{row}(M)$, si j est l'indice de la i -ème ligne de \widetilde{M} , alors $\widetilde{M}_{i,j}x^{mon_j}$ est le terme de tête du polynôme correspondant à cette ligne.*

Ces remarques justifient le nom d'algorithme LUP tropical, de même que le fait que cet algorithme calcule les termes de têtes de $Vect(Lignes(M))$. Enfin, comme les signatures restent inchangées durant le calcul de la forme LUP, il est suffisant de seulement noter les signatures de chaque ligne sur les matrices de Macaulay étiquetées.

Un algorithme F5-Matriciel utilisant les signatures

Nous montrons qu'il est possible d'appliquer le calcul de la forme LUP dans l'algorithme F5-matriciel. Ceci permettra d'utiliser le fait que lorsqu'on construit les matrices de Macaulay, les lignes déjà échelonnées ne nécessitent plus aucun travail.

Tout d'abord, le critère F5 avec signature peut être utilisé :

Proposition 9.2.24. *Soit (u, f) un polynôme homogène étiqueté de degré d . Supposons que $sign(u) = x^\alpha e_i$, avec $1 < i \leq s$ et $x^\alpha \in I_{i-1}$. Alors,*

$$x^\alpha \in Vect(\{x^\beta f_k, |x^\beta f_k| = d, \text{ et } (x^\beta, k) < (x^\alpha, i)\}).$$

En conséquence, si (u, f) est un polynôme homogène de degré d avec $sign(u) = x^\alpha e_i$ et $x^\alpha \notin LM(I_{i-1})$. Alors f peut s'écrire $f = x^\alpha f_i + g$, avec

$$g \in Vect(\{x^\beta f_k, |x^\beta f_k| = d, \text{ et } (x^\beta, k) < (x^\alpha, i)\}).$$

Un algorithme F5-Matriciel plus rapide : Nous pouvons maintenant présenter un algorithme F5-Matriciel tropical avec signatures :

Algorithme 9.2.25 : Algorithme F5-Matriciel tropical avec signatures

entrée : $F = (f_1, \dots, f_s) \in \mathcal{A}^s$, homogènes de degrés respectifs d_1, \dots, d_s , et $D \in \mathbb{N}$.

sortie : $(g_1, \dots, g_k) \in \mathcal{A}^k$, une D -base de Gröbner tropicale de $\langle F \rangle$, si D est assez grand.

début

```

   $G \leftarrow F$  ;
  pour  $d \in \llbracket 0, D \rrbracket$  faire
     $\widetilde{\mathcal{M}}_{d,0} := \emptyset$  ;
    pour  $i \in \llbracket 1, s \rrbracket$  faire
       $\mathcal{M}_{d,i} := \widetilde{\mathcal{M}}_{d,i-1}$  ;
      pour  $L$  une ligne de  $\widetilde{\mathcal{M}}_{d-1,i}$  faire
        pour  $x \in \{X_1, \dots, X_n\}$  faire
           $x^\alpha e_k := \text{sign}(xL)$  ;
          si  $k = i$ ,  $x^\alpha$  n'est pas le terme d'une ligne de  $\widetilde{\mathcal{M}}_{d-d_i,i-1}$ , et  $\mathcal{M}_{d,i}$  n'a pas déjà
          une ligne de signature  $x^\alpha e_i$  alors
            Ajouter  $xL$  à  $\mathcal{M}_{d,i}$  ;
        Calculer  $\widetilde{\mathcal{M}}_{d,i}$ , le U de la forme LUP tropicale de  $\mathcal{M}_{d,i}$  ;
        /*  $\mathcal{M}_{d,i}$  est déjà en partie sous forme échelonnée tropicale par les calculs précédents
           */
      Ajouter à  $G$  les lignes avec un nouveau monôme de tête ;
  Retourner  $G$  ;

```

Correction Cet algorithme calcule une D -base de Gröbner tropicale. La première chose à montrer est que lors de la construction des matrices de Macaulay lors de l'exécution de l'algorithme, les deux propriétés suivantes sont satisfaites : $\text{Im}(\mathcal{M}_{d,i}) = I_i \cap \mathcal{A}_d$ et pour tout monôme x^α de degré $d - d_i$ tel que $x^\alpha \notin LM(I_{i-1})$, $\mathcal{M}_{d,i}$ a une ligne ayant pour signature $x^\alpha e_i$. Ceci peut se prouver par récurrence sur d et i .

Maintenant, comme l'algorithme LUP tropical calcule une base échelonnée de $\mathcal{M}_{d,i}$, de même que lors de l'algorithme F5-Matriciel tropical précédent, l'algorithme F5-Matriciel avec signature calcule une D -base de Gröbner.

Complexité La différence principale du point de vue de la complexité entre les Algorithmes 9.2.8 et 9.2.25 est que dans le second cas, le calcul de la forme LUP tropicale de $\mathcal{M}_{d,i+1}$ prend en compte le fait que le calcul a déjà été fait pour $\mathcal{M}_{d,i}$, i.e. les premières lignes de $\mathcal{M}_{d,i+1}$ sont déjà sous forme échelonnée avec les bons termes de tête, et aucun nouveau calcul n'est à ajouter. En conséquence, la complexité du calcul d'une D -base de Gröbner tropicale de (f_1, \dots, f_s) est la même que lors de l'étude naïve du cas classique, c'est-à-dire en $O\left(sD\binom{n+D-1}{D}^3\right)$ opérations dans \mathcal{K} , pour $D \rightarrow +\infty$. Si (f_1, \dots, f_s) est une suite régulière, alors la complexité est en $O\left(D\binom{n+D-1}{D}^3\right)$.

Un exemple Nous reprenons l'Exemple 9.2.12 en lui appliquant l'algorithme 9.2.25.

Exemple 9.2.26. Nous appliquons l'Algorithme 9.2.25 sur $F = (f_1, f_2, f_3) \in \mathbb{Q}[x, y, z]$ avec valuation 2-adique, poids $w = [0, 0, 0]$ et l'ordre grevlex pour briser les égalités, et $f_1 = 2x + z$, $f_2 = x^2 + y^2 - 2z^2$ et $f_3 = 4y^2 + yz + 8z^2$. Nous prenons $D = 3$, la borne de Macaulay. Nous partons de $G = (f_1, f_2, f_3)$.

En degré 1, nous obtenons $\widetilde{\mathcal{M}}_{1,1} = \widetilde{\mathcal{M}}_{1,2} = \widetilde{\mathcal{M}}_{1,3}$ qui sont

$$f_1 \begin{vmatrix} z & y & x \\ 2 & 0 & 1 \end{vmatrix}.$$

9. Une approche tropicale

En degré 2, $\widetilde{\mathcal{M}}_{2,1}$ est

$$\begin{array}{c} z^2 \quad yz \quad xz \quad y^2 \quad xy \quad x^2 \\ \begin{array}{c} zf_1 \\ yf_1 \\ xf_1 \end{array} \left| \begin{array}{cccccc} 1 & & 2 & & & \\ & 1 & & & 2 & \\ & & 1 & & & 2 \end{array} \right| \end{array}$$

Ensuite, $\widetilde{\mathcal{M}}_{2,2}$ est

$$\begin{array}{c} z^2 \quad yz \quad xz \quad x^2 \quad xy \quad y^2 \\ \begin{array}{c} zf_1 \\ yf_1 \\ xf_1 \\ f_2 \end{array} \left| \begin{array}{cccccc} 1 & & 2 & & & \\ & 1 & & & 2 & \\ & & 1 & 2 & & \\ & & & -7 & & 1 \end{array} \right| \end{array}$$

De même : $\widetilde{\mathcal{M}}_{2,3}$ est

$$\begin{array}{c} z^2 \quad yz \quad xz \quad x^2 \quad xy \quad y^2 \\ \begin{array}{c} zf_1 \\ yf_1 \\ xf_1 \\ f_2 \\ f_3 \end{array} \left| \begin{array}{cccccc} 1 & & 2 & & & \\ & 1 & & & 2 & \\ & & 1 & 2 & & \\ & & & -7 & & 1 \\ & & & & -2 & 60/7 \end{array} \right| \end{array}$$

En degré 3, nous obtenons $\widetilde{\mathcal{M}}_{3,1}$ qui est

$$\begin{array}{c} z^3 \quad yz^2 \quad xz^2 \quad y^2z \quad xyz \quad x^2z \quad y^3 \quad xy^2 \quad x^2y \quad x^3 \\ \begin{array}{c} z^2f_1 \\ y^2f_1 \\ x^2f_1 \\ y^2f_1 \\ xyf_1 \\ x^2f_1 \end{array} \left| \begin{array}{ccccccccc} 1 & & 2 & & & & & & \\ & 1 & & & 2 & & & & \\ & & 1 & & & 2 & & & \\ & & & 1 & & & 2 & & \\ & & & & 1 & & & 2 & \\ & & & & & 1 & & & 2 \end{array} \right| \end{array}$$

la matrice suivante :

Grâce au critère F5, nous pouvons écarter zf_2 et seulement ajouter yf_2 et xf_2 pour définir $\mathcal{M}_{3,2}$.

$$\begin{array}{c} z^3 \quad yz^2 \quad xz^2 \quad y^2z \quad xyz \quad x^2z \quad x^2y \quad x^3 \quad y^3 \quad xy^2 \\ \begin{array}{c} z^2f_1 \\ y^2f_1 \\ x^2f_1 \\ y^2f_1 \\ xyf_1 \\ x^2f_1 \\ zf_2 \\ yf_2 \end{array} \left| \begin{array}{cccccccccc} 1 & & 2 & & & & & & & \\ & 1 & & & 2 & & & & & \\ & & 1 & & & 2 & & & & \\ & & & 1 & & & & & & 2 \\ & & & & 1 & & 2 & & & \\ & & & & & 1 & & 2 & & \\ & & & & & & -7 & & 1 & \\ & & & & & & & -7 & & 1 \end{array} \right| \end{array}$$

Nous obtenons ainsi $\widetilde{\mathcal{M}}_{3,2}$ comme ceci :

Enfin, à nouveau grâce au critère F5, nous pouvons écarter zf_3 et nous obtenons $\widetilde{\mathcal{M}}_{3,3}$ comme :

$$\begin{array}{c}
z^3 \quad yz^2 \quad xz^2 \quad y^2z \quad xyz \quad x^2z \quad x^2y \quad x^3 \quad xy^2 \quad y^3 \\
\begin{array}{l}
z^2f_1 \\
yzf_1 \\
xzf_1 \\
y^2f_1 \\
xyf_1 \\
x^2f_1 \\
zf_2 \\
yf_2 \\
zf_3 \\
yf_3
\end{array}
\left| \begin{array}{cccccccccc}
1 & & 2 & & & & & & & \\
& 1 & & & 2 & & & & & \\
& & 1 & & & 2 & & & & \\
& & & 1 & & & & & & 2 \\
& & & & 1 & & 2 & & & \\
& & & & & 1 & & 2 & & \\
& & & & & & -7 & & 1 & \\
& & & & & & & -7 & & 1 \\
& & & & & & & & -2 & 60/7 \\
& & & & & & & & & 1786/49
\end{array} \right.
\end{array}$$

9.3. Un algorithme FGLM tropical

Dans cette Section, nous étudions l'usage d'une base de Gröbner tropicale comme point de départ pour définir une forme normale et appliquer les algorithmes FGLM. Nous remarquons que ceux-ci s'adaptent directement et qu'ainsi, nous pouvons partir d'une base de Gröbner tropicale avec un poids $w = (0, \dots, 0)$ (dont nous avons vu précédemment que c'était le choix entraînant le moins de perte de précision par l'algèbre linéaire), pour ensuite calculer une base de Gröbner pour un ordre monomial classique par un algorithme FGLM. La première étape est de définir et calculer les matrices de multiplication.

9.3.1. Calcul des matrices de multiplication

Une première approche

Afin de calculer les matrices de multiplication dans le quotient \mathcal{A}/I , nous rappelons les définitions suivantes :

Définition 9.3.1. Soit $w \in \Gamma^n$ et \leq_1 un ordre monomial, et soit \leq l'ordre sur les termes correspondant. Nous notons $B_{\leq} = \{x^\alpha | x^\alpha \notin LM_{\leq}(I)\}$ la base, que nous appelons canonique par rapport à \leq , du \mathcal{K} -espace vectoriel \mathcal{A}/I .¹ Ses éléments sont ordonnés par degré croissant puis par ordre croissant pour \leq .

Nous définissons aussi la forme normale d'un polynôme, ou projeté sur \mathcal{A}/I .

Définition 9.3.2. Soit $P \in \mathcal{A}$. Nous définissons $NF_{\leq}(P)$, forme normale de P pour I , comme le projeté de P dans \mathcal{A}/I (ou encore le reste de $P \bmod I$) écrit dans la base B_{\leq} .

Afin de calculer aisément dans \mathcal{A}/I et de calculer aisément les formes normales, nous introduisons à nouveau les matrices de multiplication par les variables de \mathcal{A} . Elles permettront à nouveau de calculer la projection $NF_{\leq}(P)$ d'un $P \in \mathcal{A}$ par de simples multiplications matrices-vecteurs et additions.

Définition 9.3.3 (Matrices de multiplication). Pour $i \in \llbracket 1, n \rrbracket$, nous notons T_i , la matrice de la multiplication par x_i dans \mathcal{A}/I écrite dans la base B_{\leq} . Nous l'appelons i -ème matrice de multiplication.

La caractérisation donnée pour le bord de I dans [FGLM93] et la Proposition 6.2.33 sont encore vérifiées. Néanmoins, ceci n'est pas suffisant pour appliquer à nouveau l'Algorithme 6.2.34. En effet, dans le troisième cas de la boucle **pour** de cet algorithme (le second **sinon**), il n'est *a priori* pas vrai que pour tous les monômes x^β de $NF_{\leq}(v)$ on a déjà calculé $T_i x^\beta$.

1. Elle est bien sûr génératrice, et libre puisque toute relation donnerait un nouveau terme de tête pour I .

Matrices de Macaulay étiquetées échelonnées réduites tropicalement

Néanmoins, nous allons pouvoir calculer les matrices de multiplication à partir des matrices de Macaulay étiquetées échelonnées réduites tropicalement jusqu'au degré D , avec D plus petit degré d tel que $I \cap \mathcal{A}_d = \mathcal{A}_d$.

Définition 9.3.4. Soit (M, mon) une matrice de Macaulay étiquetée. Elle est dite sous forme échelonnée réduite tropicale si :

- pour chaque ligne, le premier coefficient non nul correspond au monôme de tête du polynôme correspondant à cette ligne ;
- pour chaque monôme apparaissant comme monôme de tête d'une ligne, il n'y a qu'une seule ligne ayant un coefficient non-nul pour la colonne correspondant à ce monôme.

Supposons que $M \in M_{k,l}(K)$ et soit (M', mon) avec $M' \in M_{k,l}(\mathcal{K})$. (M', mon) est dite être une forme échelonnée réduite tropicale de (M, mon) si (M', mon) est sous forme échelonnée réduite et il existe $P \in GL_k(\mathcal{K})$ tel que $M' = PM$.

L'algorithme suivant calcule une forme échelonnée réduite à partir du U de la forme LUP ou à partir de la forme échelonnée tropicale d'une matrice de Macaulay étiquetée.

Algorithme 9.3.5 : Calcul d'une forme échelonnée réduite tropicale

entrée : (M, mon) une matrice de Macaulay, résultat du calcul du U d'une forme LUP ou de la forme échelonnée tropicale d'une matrice de Macaulay étiquetée.
sortie : (M', mon) , une forme échelonnée réduite tropicale de (M, mon) .

début

```

     $n_{row} :=$  le nombre de lignes de  $M$  ;
     $n_{col} :=$  le nombre de colonnes de  $M$  ;
    pour  $j$  de  $n_{col}$  à 1 faire
        si il existe  $i$  tel que le premier coefficient non nul de la ligne  $i$  est celui de la colonne  $j$ 
        alors
            En pivotant avec la ligne  $i$ , éliminer tous les coefficients non nuls hors de la ligne  $i$ 
            sur la colonne  $j$  ;

```

Proposition 9.3.6. Soit (M, mon) une matrice de Macaulay, résultat du calcul du U d'une forme LUP ou de la forme échelonnée tropicale d'une matrice de Macaulay étiquetée. Alors l'Algorithme 9.3.5 calcule une forme échelonnée réduite tropicale de (M, mon) . La complexité est en $r \times n_{col}^2$ opérations arithmétiques dans \mathcal{K} , avec r le rang de M et n_{col} son nombre de colonnes.

Si les coefficients de M sont connus à précision finie, alors la perte de précision est la valuation du produit des pivots.

Démonstration. Le résultat est clair, vu par exemple la démonstration du Théorème 1.2.6. □

Exemple 9.3.7. Nous reprenons les matrices apparaissant dans l'exemple 9.2.12 avec le cadre de la précision finie de l'exemple 9.2.17. En particulier, nous avons $\widetilde{\mathcal{M}}_{2,3}$ qui est :

$$\begin{array}{c}
 \begin{array}{ccccc}
 & x^2 & xz & yz & z^2 & xy & y^2 \\
 zf_1 & \left| \begin{array}{cccc} & & & 1017 & & 4 \end{array} \right| & +O(2^{10}) \\
 yf_1 & \left| \begin{array}{cccc} & & 1 & & 2 & \end{array} \right| & +O(2^{10}) \\
 xf_1 & \left| \begin{array}{cccc} & 1 & & 4 & & 1022 \end{array} \right| & +O(2^{10}) \\
 f_2 & \left| \begin{array}{cccc} 1 & & & 1022 & & 1 \end{array} \right| & +O(2^{10}) \\
 f_3 & \left| \begin{array}{cccc} & & & & 1022 & 740 \end{array} \right| & +O(2^{10})
 \end{array}
 \end{array}$$

Alors, après application de l'Algorithme 9.3.5, nous obtenons la matrice de Macaulay réduite

suivante :

| | x^2 | xz | yz | z^2 | xy | y^2 | |
|--------|-------|------|------|-------|------|-------|--------------|
| zf_1 | | | | 1017 | | 4 | $+O(2^{10})$ |
| yf_1 | | | 1 | | | 740 | $+O(2^{10})$ |
| xf_1 | | 1 | | | | 878 | $+O(2^{10})$ |
| f_2 | 1 | | | | | 585 | $+O(2^{10})$ |
| f_3 | | | | 1022 | 740 | | $+O(2^{10})$ |

Calcul des matrices de multiplication par les matrices de Macaulay

Maintenant, étant données les matrices de Macaulay échelonnées réduites tropicalement pour un système de polynômes donné engendrant un idéal de dimension zéro, nous avons l'algorithme suivant pour calculer les matrices de multiplication :

Algorithme 9.3.8 : Calcul des matrices de multiplication dans le cas tropical

entrée : $F = (f_1, \dots, f_s)$ engendrant un idéal $I \subset \mathcal{A}$ de dimension zéro et de degré δ pour un ordre sur les termes \leq . G une base de Gröbner tropicale de I . $B_{\leq} = (\epsilon_1, \epsilon_2 \leq \dots, \epsilon_\delta)$ la base canonique de \mathcal{A}/I pour \leq . Les matrices de Macaulay étiquetées échelonnées réduites (tropicalement) $\widetilde{Mac}_d(F)$ pour d de 1 jusqu'à D avec D tel que $I \cap \mathcal{A}_D = \mathcal{A}_D$.
sortie : Les matrices de multiplication T_i pour I et \leq .

début

```

pour  $i \in \llbracket 1, n \rrbracket$  faire
   $T_i := 0_{\delta \times \delta}$  ;
   $L := [x_i \epsilon_k | i \in \llbracket 1, n \rrbracket \text{ et } \epsilon_k \in B_{\leq}]$ , triée par degré puis par ordre croissant, sans répétition ;
  pour  $d$  de 1 à  $D - 1$  faire
    Soit  $mon_d$  la liste des monômes correspondant aux colonnes de  $\widetilde{Mac}_d(F)$  ;
    pour  $u \in mon_d$  en partant du monôme correspondant à la dernière colonne de  $\widetilde{Mac}_d(F)$  faire
      si  $u \in \mathcal{E}_{\leq}(I)$  alors
         $T_i[u, u/x_i] := 1$  pour tout  $i$  tel que  $x_i | u$  et  $x^\alpha \in \mathcal{E}_{\leq}(I) \cap \mathcal{A}_d$  ;
      sinon si  $u \in L$  alors
        La seule ligne de  $\widetilde{Mac}_d(F)$  de monôme de tête  $u$  correspond au polynôme
         $cu + \sum_{\alpha} c_{\alpha} x^{\alpha}$ , avec les  $x^{\alpha}$  qui sont dans  $\mathcal{E}_{\leq}(I) \cap \mathcal{A}_d$  ;
         $T_i[x^{\alpha}, u/x_i] := \frac{c_{\alpha}}{c}$  pour tout  $i$  tel que  $x_i | u$  et  $x^{\alpha} \in \mathcal{E}_{\leq}(I) \cap \mathcal{A}_d$  ;
      fin
    pour  $u \in L \cap \mathcal{A}_D$  faire
       $T_i[\cdot, u/x_i] := 0$  pour tout  $i$  tel que  $x_i | u$  ;
    Retourner  $T_1, \dots, T_n$  ;

```

Proposition 9.3.9. Soit $F = (f_1, \dots, f_s)$ engendrant un idéal $I \subset \mathcal{A}$ de dimension zéro et de degré δ et \leq un ordre sur les termes. Soit G une base de Gröbner tropicale de I , et nous supposons données les matrices de Macaulay étiquetées échelonnées réduites (tropicalement) $\widetilde{Mac}_d(F)$ pour d de 1 jusqu'à D avec D tel que $I \cap \mathcal{A}_D = \mathcal{A}_D$. Alors, l'Algorithme 9.3.8 calcule les matrices de multiplication de \mathcal{A}/I pour \leq . Ceci est fait sans opération arithmétique.

Démonstration. Pour les éléments de $L \cap \mathcal{A}_{<D}$, tout est clair avec la définition de matrices de Macaulay étiquetées échelonnées réduites (tropicalement), définition 9.3.4. Pour les éléments de $L \cap \mathcal{A}_{\geq D}$, comme $I \cap \mathcal{A}_{\geq D} = \mathcal{A}_{\geq D}$, leur forme normale est nulle, d'où le résultat. \square

9.3.2. Application aux algorithmes FGLM

Une présentation d'un algorithme FGLM tropical

À partir du calcul des matrices de multiplication, nous pouvons travailler dans le quotient avec l'Algorithme 6.2.30, et alors, toutes les variantes de l'algorithme FGLM sont disponibles directement. Nous présentons une manière de procéder avec l'algorithme suivant :

Algorithme 9.3.10 : Algorithmes FGLM tropicaux

entrée : $F = (f_1, \dots, f_s)$ engendrant un idéal $I \subset \mathcal{A}$ de dimension zéro et de degré δ pour un ordre sur les termes \leq . D une majoration du degré de régularité de I . \leq_2 un ordre monomial.

sortie : Une base de Gröbner G pour \leq_2 de l'idéal I .

début

Appliquer l'Algorithme F5-Matriciel tropical avec signature (9.2.25) pour F , \leq et D ;
Calculer avec l'Algorithme 9.3.5 des formes échelonnées réduites tropicales des matrices de Macaulay considérées lors de l'étape précédente ;

À partir de ces matrices de Macaulay sous forme échelonnée réduite tropicale, calculer T_1, \dots, T_n les matrices de multiplication de \mathcal{A}/I pour \leq avec l'Algorithme 9.3.8 ;

si \leq_2 est un ordre *lex* et que I est en position générale **alors**

Appliquer l'Algorithme 6.2.48 avec les matrices de multiplication T_1, \dots, T_n pour obtenir G base de Gröbner de I pour \leq_2 ;

sinon

Appliquer l'Algorithme 6.2.38 avec les matrices de multiplication T_1, \dots, T_n pour obtenir G base de Gröbner de I pour \leq_2 ;

Le comportement de cet algorithme est alors donné par la proposition suivante :

Proposition 9.3.11. *Soit $F = (f_1, \dots, f_s)$ engendrant un idéal $I \subset \mathcal{A}$ de dimension zéro et de degré δ pour un ordre sur les termes \leq et \leq_2 un ordre monomial. Alors l'Algorithme 9.3.10 calcule une base de Gröbner G de I pour \leq_2 . Le temps de calcul en nombre d'opérations dans \mathcal{K} se décompose ainsi :*

- $O\left(sD\binom{n+D-1}{D}^3\right)$ pour l'algorithme F5-Matriciel avec signature plus le calcul de la forme échelonnée réduite tropicale, ou $O\left(D\binom{n+D-1}{D}^3\right)$ dans le cas d'une suite régulière ;
- $O(1)$ pour le calcul des matrices de multiplication ;
- $O(n\delta^3)$ pour l'algorithme FGLM, ou $O(\delta^3 + n\delta^2)$ dans le cas d'un idéal en position générale.

Étude à précision finie

Le cadre précédent s'adapte très bien à la précision finie en remplaçant les algorithmes utilisés par leur version stabilisée. Nous présentons avec l'algorithme suivant une manière de procéder :

Algorithme 9.3.12 : Algorithmes FGLM tropicaux numériques

entrée : $F = (f_1, \dots, f_s)$ engendrant un idéal $I \subset A$ de dimension zéro et de degré δ pour un ordre sur les termes \leq . D une majoration du degré de régularité de I . \leq_2 un ordre monomial.

sortie : Une base de Gröbner G pour \leq_2 de l'idéal I .

début

Appliquer l'Algorithme F5-Matriciel tropical (9.2.8) pour F , \leq et D ;
 Calculer avec l'Algorithme 9.3.5 des formes échelonnées réduites tropicales des matrices de Macaulay considérées lors de l'étape précédente ;
 À partir de ces matrices de Macaulay sous forme échelonnée réduite tropicale, calculer T_1, \dots, T_n les matrices de multiplication de A/I pour \leq avec l'Algorithme 9.3.8 ;
si \leq_2 est un ordre lex et que I est en position générale **alors**
 Appliquer l'Algorithme 8.2.1 avec les matrices de multiplication T_1, \dots, T_n pour obtenir G base de Gröbner de I pour \leq_2 ;
sinon
 Appliquer l'Algorithme 8.1.1 avec les matrices de multiplication T_1, \dots, T_n pour obtenir G base de Gröbner de I pour \leq_2 ;

La définition de conditionnement pour le changement d'ordre vu en Définition 8.1.4 s'étend très naturellement au cas tropical. Nous avons alors le résultat suivant :

Proposition 9.3.13. Soit $F = (f_1, \dots, f_n)$ une suite régulière engendrant un idéal $I \subset A$ de dimension zéro et de degré δ pour un ordre sur les termes \leq et \leq_2 un ordre monomial. Alors, si la précision sur les coefficients des polynômes de F est suffisante, l'Algorithme 9.3.12 calcule une base de Gröbner G de I pour \leq_2 . Le temps de calcul en opérations sur K à la précision initiale se décompose ainsi :

- $O\left(nD\binom{n+D-1}{D}^3\right)$ pour l'algorithme F5-Matriciel tropical plus le calcul de la forme échelonnée réduite tropicale ;
- $O(1)$ pour le calcul des matrices de multiplication ;
- $O(n\delta^3)$ pour l'algorithme FGLM, ou $O(\delta^3 + n\delta^2)$ dans le cas d'un idéal en position générale.

Proposition 9.3.14. Dans le cadre de la proposition précédente, une précision suffisante est donnée par :

$$prec_{MF5trop}(F, D, \leq) + loss_{MF5trop}(F, D, \leq) + cond_{\leq, \leq_2}^+.$$

La perte de précision est alors majorée par :

$$2 \times loss_{MF5trop}(F, D, \leq) + 2cond_{\leq, \leq_2}.$$

Démonstration. L'étude de la précision est donnée ainsi :

- besoin de $prec_{MF5trop}(F, D, \leq)$ pour exécuter l'algorithme F5-Matriciel tropicale, et perte de $loss_{MF5trop}(F, D, \leq)$ lors de son exécution ;
- perte de $loss_{MF5trop}(F, D, \leq)$ pour le calcul des formes échelonnées réduites tropicales des matrices de Macaulay ;
- perte de $2cond_{w+\leq, \leq_2}$ pour l'application de l'algorithme FGLM stabilisé. Nous nous restreignons à utiliser $cond_{\leq, \leq_2}^+$ car les éventuels gains de précision ne peuvent compenser une trop faible précision en entrée de la partie FGLM de cet algorithme.

□

Un exemple

Exemple 9.3.15. Nous appliquons l'Algorithme FGLM tropical 9.3.12 sur la famille de polynôme de l'exemple 7.2.15.

9. Une approche tropicale

Celle-ci est donnée par $F = (f_1, f_2, f_3) \in \mathbb{Q}_2[x, y, z]$ avec les f_i connus à précision $O(2^{10})$: $f_1 = (2 + O(2^{10}))x + (1 + O(2^{10}))z$, $f_2 = (1 + O(2^{10}))x^2 + (1 + O(2^{10}))y^2 - (2 + O(2^{10}))z^2$ et $f_3 = (4 + O(2^{10}))y^2 + (1 + O(2^{10}))yz + (8 + O(2^{10}))z^2$. Nous prenons $D = 3$, la borne de Macaulay.

Nous calculons d'abord une base de Gröbner tropicale réduite pour $w = (0, 0, 0)$ et grevlex avec l'Algorithme 9.2.8, et obtenons notamment comme premier escalier $[1, x, y, y^2]$. Nous appliquons maintenant l'Algorithme FGLM tropical numérique 9.3.12 à partir des matrices de Macaulay réduites obtenues pour en déduire une base de Gröbner pour l'ordre lexicographique avec $z > y > x$. Nous obtenons la matrice suivante comme forme normale de Smith de la matrice de changement de base :

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Le nouvel escalier est $[1, x, x^2, y]$. Nous obtenons alors comme base de Gröbner (rendue minimale) en sortie :

$$(x^3, xy + (482 + O(2^9))x^2, y^2 + (1017 + O(2^{10}))x^2, z + (2 + O(2^{10}))x, x^2z).$$

Remarquons que par rapport au calcul mené lors de l'exemple 8.1.13, nous obtenons deux chiffres (en base 2) de plus sur le coefficient en x^2 du troisième polynôme.

9.4. Méthode tropicale pour des calculs de bases de Gröbner classiques

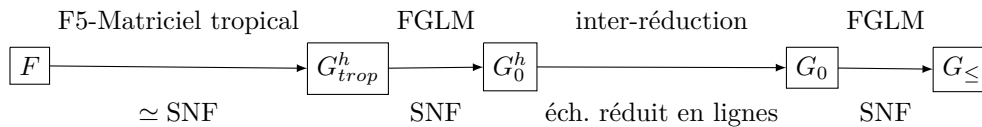
Grâce aux résultats précédents, nous pouvons conclure quant au calcul de bases de Gröbner en dimension zéro avec le résultat suivant :

Théorème 9.4.1. *Soit $F = (f_1, \dots, f_n)$ des polynômes de $A = \mathbb{Q}_p[X_1, \dots, X_n]$ tels que leurs composantes homogènes de plus haut degré $F^h = (f_1^h, \dots, f_n^h)$ forme une suite régulière de polynômes homogènes de $A = \mathbb{Q}_p[X_1, \dots, X_n]$. Soit $I^h = \langle F^h \rangle$. Soit δ le degré de l'idéal engendré par F . Soit \leq un ordre monomial sur A . Supposons que \leq ne raffine pas le degré. Alors si la précision sur les coefficients des f_i est assez grande, il est possible de calculer une base de Gröbner de F en suivant la méthode suivante.*

1. Calculer une base de Gröbner G_{trop}^h de $\langle F^h \rangle$ pour l'ordre tropical donné par $w = (0, \dots, 0)$ et grevlex grâce à un algorithme F5-Matriciel tropical. Obtenir l'écriture des éléments de G_{trop}^h en fonction de ceux de F^h ;
2. Calculer une base de Gröbner G_0^h pour grevlex de $\langle F^h \rangle$ grâce à G_{trop}^h et un algorithme FGLM ;
3. Grâce à l'écriture des éléments de G_0^h en fonction de ceux de F^h ³, en déduire une base de Gröbner G_0 de $\langle F \rangle$ pour grevlex ;
4. Grâce à l'algorithme FGLM, en déduire une base de Gröbner de $\langle F \rangle$ pour \leq .

Lorsque \leq raffine le degré, il est possible de remplacer grevlex par \leq et de s'arrêter à l'étape 3.

Ceci correspond au schéma suivant :



Proposition 9.4.2. *Dans le contexte du théorème précédent, le temps de calcul est en*

$$O\left(n^2 D \binom{n+D-1}{D}^3\right) + O(n\delta^3)$$

2. Nous avons utilisé l'écriture en base 10 plutôt qu'en base 2 par concision.

3. Ceci peut s'obtenir grâce aux polynômes G_{trop}^h et leur écriture en fonction de F^h .

opérations dans K à la précision initiale (en négligeant les éventuels gains de précision). La précision nécessaire est majorée par

$$\begin{aligned} \text{prec}_{MF5trop}(F^h, D, \leq) + \text{loss}_{MF5trop}(F^h, D, \leq) + 2\text{cond}_{w+\text{grevlex}, \text{grevlex}}^+(I^h) \\ + \text{cond}(G_0, \text{grevlex}) + \text{cond}_{\text{grevlex}, \leq}^+(I). \end{aligned}$$

La perte de précision est alors majorée par :

$$2\text{loss}_{MF5trop}(F^h, D, \leq) + 2\text{cond}_{w+\text{grevlex}, \text{grevlex}}^+(I^h) + \text{cond}(G_0, \text{grevlex}) + 2\text{cond}_{\text{grevlex}, \leq}^+(I).$$

Démonstration. Avec ce qui précède, tout est clair hormis l'étape 3 du théorème et l'estimation du temps de calcul. Autrement dit, il reste à expliquer comment écrire les éléments de G_0^h en fonction de ceux de F^h et quel est le sur-coût de ce calcul par rapport à un algorithme F5-Matriciel classique.

L'idée est la suivante : lors du calcul de G_{trop}^h , on utilise des matrices de Macaulay tropicales avec étiquettes. Autrement dit, à chaque ligne d'une matrice de Macaulay est attachée une étiquette, l'écriture du polynôme correspondant à cette ligne en fonction de F . Lorsqu'on construit la matrice, il suffit d'attacher à la ligne $x^\alpha f_i$ le n -uplet $(0, \dots, 0, x^\alpha, 0, \dots, 0)$ (x^α en i -ème position). Ensuite, on répercute les opérations effectuées sur les lignes sur leurs étiquettes.

En sortie de l'algorithme F5-Matriciel tropical, on obtient bien ainsi l'écriture de G_{trop}^h en fonction de F^h . Notons Macred les matrices de Macaulay échelonnées réduites tropicales, avec étiquettes, utilisées lors du calcul.

Maintenant, une fois calculée G_0^h base de Gröbner de I^h pour grevlex grâce à l'Algorithme FGLM stabilisé, nous écrivons les éléments de G_0^h en fonction de F^h . Pour cela, il suffit de réduire les éléments de G_0^h par les matrices de Macaulay réduites Macred. Une réduction à la manière d'un échelonnement en ligne suffit. Il n'y a pas de problème de test à zéro : si $g^h \in G_0^h$, une fois éliminé les monômes de g^h qui sont monôme de tête d'une ligne de la matrice de Macaulay réduite du même degré Macred $_{|g|}$, le fait que $g^h \in I^h$ et que l'on connaisse tous les monômes pouvant apparaître comme monôme de tête de $I^h \cap A_{|g^h|}$ implique que le reste est nul. En répercutant la réduction effectuée sur les étiquettes des lignes de Macred $_{|g^h|}$, on obtient l'écriture $g^h = \sum_{i=1}^n a_{g^h, i} f_i^h$.

En conséquence nous pouvons écrire $g = \sum_{i=1}^n a_{g^h, i} f_i \in I$. Ceci nous permet de définir G_0 qui a les mêmes termes de tête que G_0^h . Grâce au Théorème 7.6.1, $LM_{\text{grevlex}}(I) = LM_{\text{grevlex}}(I^h)$, et donc G_0 est bien une base de Gröbner de I pour grevlex.

Concernant le temps de calcul, il suffit d'estimer le coût du travail avec des étiquettes lors du calcul de G_{trop}^h . Ce coût peut être majoré par celui qui consisterait à travailler avec des matrices de Macaulay ayant $(s+1)$ fois plus de colonnes. Ceci explique le facteur supplémentaire s par rapport à l'estimation de complexité en Sous-Sous-Section 9.2.1. □

Ce résultat montre en particulier la continuité (et même la différentiabilité) du calcul de bases de Gröbner en dimension zéro au voisinage de polynômes dont les composantes homogènes de plus haut degré forment une suite régulière, condition qui définit un ouvert de Zariski non vide.

Exemple 9.4.3. Nous appliquons la méthode précédente sur une famille de polynôme dont les composantes homogènes de plus haut degré sont celles de l'exemple 7.2.15.

Notre famille est donnée par $F = (f_1, f_2, f_3) \in \mathbb{Q}_2[x, y, z]$ avec les f_i connus à précision $O(2^{10})$: $f_1 = (2 + O(2^{10}))x + (1 + O(2^{10}))z + 1$, $f_2 = (1 + O(2^{10}))x^2 + (1 + O(2^{10}))y^2 - (2 + O(2^{10}))z^2 + x + z$ et $f_3 = (4 + O(2^{10}))y^2 + (1 + O(2^{10}))yz + (8 + O(2^{10}))z^2 + y - z + 2$. Nous souhaitons calculer une base de Gröbner de $\langle f_1, f_2, f_3 \rangle$ pour l'ordre lexicographique $z > y > x$.⁴

Nous prenons $D = 3$, la borne de Macaulay. Nous calculons d'abord une base de Gröbner tropicale réduite pour $w = (0, 0, 0)$ et grevlex avec l'Algorithme 9.2.8 sur les composantes homogènes de plus haut degré, F^h et obtenons notamment comme premier escalier $[1, x, y, y^2]$. Nous n'avons pas constaté de perte de précision sur cette étape.

Nous appliquons ensuite l'Algorithme FGLM tropical numérique 9.3.12 à partir des matrices de Macaulay réduites obtenues pour en déduire une base de Gröbner pour l'ordre grevlex avec

4. Nous n'avons pas pris $x > y > z$ pour avoir des calculs plus intéressants dans FGLM.

9. Une approche tropicale

$x > y > z$. Nous obtenons la matrice suivante comme forme normale de Smith de la matrice de changement de base :

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix}.$$

Le nouvel escalier est $[1, z, y, z^2]$. Nous obtenons alors comme base de Gröbner (rendue minimale) en sortie pour F^h :

$$\begin{aligned} & x + (2^{-1} + O(2^8))z, \\ & yz + (15 + O(2^8))z^2, \\ & y^2 + \left(\frac{505}{4} + O(2^7)\right)z^2, \\ & \quad z^3. \end{aligned}$$

Nous en déduisons alors la base de Gröbner réduite pour $\langle F \rangle$ pour grevlex :

$$\begin{aligned} & x + (2^{-1} + O(2^8))z + 2^{-1} + O(2^8), \\ & yz + (15 + O(2^8))z^2 + y + (251 + O(2^8))z + 3 + O(2^8), \\ & y^2 + \left(\frac{505}{4} + O(2^7)\right)z^2 + z + \frac{511}{4} + O(2^7), \\ & z^3 + (117 + O(2^9))z^2 + (884 + O(2^{10}))y + 19z + 123 + O(2^9). \end{aligned}$$

Enfin, nous appliquons l'Algorithme 8.1.1 pour en déduire une base de Gröbner de $\langle F \rangle$ pour lex ($z > y > x$). La forme normale de Smith de la matrice de changement de base est :

$$\begin{bmatrix} 2^{-3} & & & \\ & 2^{-2} & & \\ & & 1 & \\ & & & 2^2 \end{bmatrix}.$$

Le résultat final est :

$$\begin{aligned} & x^4 + x^3 + \left(\frac{437}{4} + O(2^7)\right)x^2 + (52 + O(2^6))x + \frac{35}{2} + O(2^5), \\ & y + (125 + O(2^8))x^3 + (125 + O(2^8))x^2 + (381/4 + O(2^7))x, \\ & \quad z + (2 + O(2^{10}))x + 1 + O(2^9). \end{aligned}$$

9.5. Implémentation

Une implémentation jouet en Sage [S⁺11] des algorithmes précédents est disponible sur <http://perso.univ-rennes1.fr/tristan.vaccon/fglm.sage>. Comme le but de cette implémentation est l'étude de la précision, elle n'est pas nécessairement optimisée pour ce qui est du temps de calcul. Pour les lignes renseignant *non* dans la colonne *affine*, nous avons appliqué l'algorithme F5-Matriciel tropical à des polynômes homogènes de degrés donnés pour $w = (0, \dots, 0)$ et grevlex pour briser les égalités. Nous avons ensuite appliqué un algorithme FGLM pour en déduire une base de Gröbner pour l'ordre lexicographique. Nous avons pris ces polynômes homogènes en tirant des coefficients aléatoirement dans \mathbb{Z}_p (pour la mesure de Haar) : nous obtenons f_1, \dots, f_s , de degrés d_1, \dots, d_s dans $\mathbb{Z}_p[X_1, \dots, X_s]$, connus à précision $O(p^{prec})$. Nous posons D la borne de Macaulay. Cette expérience est réalisé nb_{test} fois pour chaque choix de paramètres et sont notés la perte de précision maximale (hors échec), moyenne (hors échec), et le nombre d'échecs. Ce dernier apparait comme un couple où la première composante est le nombre d'échecs pour la partie F5-Matriciel tropical et la seconde composante pour la partie FGLM.

Pour les autres lignes, nous avons fait les expériences avec les même paramètres pour des polynômes non-supposés homogènes avec bornes d_1, \dots, d_s sur les degrés. Pour ceux-ci, nous avons appliqué un algorithme F5-Matriciel Tropical sur leur partie homogène ($w = (0, \dots, 0)$ et grevlex pour briser les égalités), puis un algorithme FGLM vers grevlex sur le résultat pour en déduire une base de Gröbner pour grevlex de l'idéal engendré par les polynômes initiaux. Enfin, nous appliquons un dernier algorithme FGLM pour obtenir cette fois une base pour l'ordre lexicographique. Le nombre d'échecs est cette fois-ci représenté par un triplet, où chaque composante représente le

nombre d'échec à l'une des trois étapes du calcul. Les résultats sont consignés dans les tableaux suivant :

| $d =$ | nb_{test} | affine | D | p | $prec$ | perte maximale | perte moyenne | échecs |
|---------|-------------|--------|---|-----|--------|----------------|---------------|---------|
| [2,3,3] | 50 | non | 6 | 2 | 50 | 13 | 1 | (0,0) |
| [2,3,3] | 50 | non | 6 | 7 | 50 | 4 | 0,2 | (0,0) |
| [2,3,3] | 50 | oui | 6 | 2 | 100 | 142 | 71 | (0,0,0) |
| [2,3,3] | 20 | oui | 6 | 7 | 100 | 85 | 17 | (0,0,0) |
| [3,3,3] | 20 | non | 7 | 2 | 50 | 17 | 1 | (0,0) |
| [3,3,3] | 20 | non | 7 | 7 | 50 | 4 | 0,2 | (0,0) |
| [3,3,3] | 20 | oui | 7 | 2 | 100 | 115 | 51 | (0,0,1) |
| [3,3,3] | 20 | oui | 7 | 7 | 100 | 49 | 20 | (0,0,1) |
| [3,3,4] | 20 | non | 8 | 2 | 50 | 15 | 2 | (0,0) |
| [3,3,4] | 20 | non | 8 | 7 | 50 | 4 | 0,3 | (0,0) |
| [3,3,4] | 20 | oui | 8 | 2 | 100 | 156 | 60 | (0,0,0) |
| [3,3,4] | 20 | oui | 8 | 7 | 100 | 98 | 30 | (0,0,0) |

| $d =$ | nb_{test} | affine | D | p | $prec$ | perte maximale | perte moyenne | échecs |
|---------|-------------|--------|---|-------|--------|----------------|---------------|---------|
| [2,2,2] | 50 | non | 6 | 2 | 50 | 21 | 1 | (0,0) |
| [2,2,2] | 50 | non | 6 | 7 | 50 | 6 | 0,2 | (0,0) |
| [2,2,2] | 50 | non | 6 | 65519 | 50 | 0 | 0 | (0,0) |
| [2,2,2] | 50 | oui | 6 | 2 | 100 | 62 | 14 | (0,0,0) |
| [2,2,2] | 20 | oui | 6 | 7 | 100 | 24 | 2,7 | (0,0,0) |
| [2,2,2] | 20 | oui | 6 | 65519 | 100 | 0 | 0 | (0,0,0) |
| [3,3,3] | 20 | non | 7 | 2 | 50 | 18 | 2 | (0,0) |
| [3,3,3] | 20 | non | 7 | 7 | 50 | 2 | 0,2 | (0,0) |
| [3,3,3] | 20 | non | 7 | 65519 | 50 | 0 | 0 | (0,0) |
| [3,3,3] | 20 | oui | 7 | 2 | 100 | 101 | 55 | (0,0,3) |
| [3,3,3] | 20 | oui | 7 | 7 | 100 | 89 | 16 | (0,0,0) |
| [3,3,3] | 20 | oui | 7 | 65519 | 100 | 0 | 0 | (0,0,0) |
| [4,4,4] | 20 | non | 8 | 2 | 50 | 22 | 2,2 | (0,0) |
| [4,4,4] | 20 | non | 8 | 7 | 50 | 4 | 0,2 | (0,0) |
| [4,4,4] | 20 | non | 8 | 65519 | 50 | 0 | 0 | (0,0) |
| [4,4,4] | 20 | oui | 8 | 2 | 100 | 100 | 76 | (0,0,9) |
| [4,4,4] | 20 | oui | 8 | 7 | 100 | 99 | 43 | (0,0,0) |
| [4,4,4] | 20 | oui | 8 | 65519 | 100 | 0 | 0 | (0,0,0) |

Nous pouvons remarquer que ces résultats suggèrent, là encore, une différence d'ordre de grandeur sur la perte de précision entre le cas homogène et le cas affine.

Nous remarquons aussi que passer par un ordre tropical permet, à précision donnée, plus de calculs de bases de Gröbner menés à terme que de passer directement par grevlex et lex, dans le cas homogène comme dans le cas affine.

Enfin, il est clair la perte de précision décroît avec le choix de p : sur des petites instances comme ici, $p = 65519$ rend les pertes de précision très peu probables.

Perspectives et questions ouvertes

“Why do you build, knowing
destruction is inevitable ? Why do
you yearn to live, knowing all things
must die ?”

Kefka *Final Fantasy VI*

"So much universe, so little time"

Terry Pratchett, *The Last Hero*

Dans le prolongement des travaux de cette thèse, diverses directions et questions apparaissent naturellement :

1. Le Chapitre 3 peut se poursuivre avec une étude plus approfondie de la précision différentielle pour des opérations sur les polynômes. De même, de nombreux problèmes en algèbre linéaire restent à traiter, comme par exemple la recherche de valeurs propres.
2. Un projet intéressant serait de combiner l’usage des réseaux comme modèle de précision, notre lemme principal 2.2.4 et la différentiation automatique. Ceci permettrait d’avoir, même dans un contexte où l’on ne comprend pas bien le calcul que l’on souhaite faire, une idée de la différentielle et ainsi, du comportement de la précision.
3. Notre étude des équations différentielles dans le Chapitre 5 ne permet pas de traiter l’équation $y'^2 = g(x)h(y)$ dans le cas $p = 2$. Numériquement, il semble cependant que pour de telles équations différentielles issues du calcul d’isogénies normalisées entre courbes elliptiques, le comportement pour $p = 2$ de la précision est le même que dans le cas général. Ceci n’est pas directement apparent sur la différentielle calculée. Ainsi, comprendre ce comportement reste un problème, à notre connaissance, ouvert.
4. Nous n’avons pas pu utiliser toute la puissance des variantes de l’algorithme FGLM développées dans [FM11, FM13, Mou13, FGHR13, FGHR14, Huo13] pour le cas de la précision finie lors de notre étude au Chapitre 8. Il serait intéressant de voir lesquelles peuvent être adaptées, et pour quel effet sur la précision et la complexité.
5. Comparer les bases de Gröbner tropicales étudiées au Chapitre 9 et les bases de bord semble naturel. Étudier le comportement de la précision lors du calcul d’une base de bord dans un contexte p -adique, et comparer la perte de précision avec celle du calcul d’une base de Gröbner (éventuellement tropicale) apporterait certainement quelque chose à ces deux théories.
6. De la même manière, il serait certainement intéressant de comparer plus précisément la méthode TSV (voir [FL11a]) avec les méthodes tropicales.
7. Une extension naturelle à l’étude de l’algorithme F5-Matriciel dans le cas tropical effectuée au Chapitre 9 serait d’adapter l’algorithme F5. Ceci permettrait certainement une implémentation efficace du point de vue du temps de calcul du calcul de bases de Gröbner tropicales.

Bibliographie

- [AKR09] Timothy G. Abbott, Kiran S. Kedlaya, and David Roe. Bounding Picard numbers of surfaces using p -adic cohomology. In *Arithmetics, geometry, and coding theory (AGCT 2005)*, pages 125–159, 2009.
- [Arn03] Elizabeth A. Arnold. Modular Algorithms for Computing Gröbner Bases. *J. Symb. Comput.*, 35(4) :403–419, April 2003.
- [Bar04] Magali Bardet. Étude des systèmes algébriques surdéterminés. applications aux codes correcteurs et à la cryptographie. *Thèse de doctorat, Université Paris VI*, 2004.
- [BBB⁺13] Christian Batut, Karim Belabas, Dominique Benardi, Henri Cohen, and Michel Olivier. *User’s guide to PARI-GP*, 1985–2013.
- [BCP97] Wieb Bosma, John Cannon, and Catherine Payoust. The Magma algebra system. I. The user language. *J. Symbolic Comput.*, 24(3-4) :235–265, 1997.
- [Ber84] Stuart J. Berkowitz. On computing the determinant in small parallel time using a small number of processors. *Information Processing Letters*, 18(3) :147 – 150, 1984.
- [BFS14] Magali Bardet, Jean-Charles Faugère, and Bruno Salvy. On the Complexity of the F5 Gröbner basis Algorithm. *Journal of Symbolic Computation*, pages 1–24, September 2014. 24 pages.
- [BGVPS05] Alin Bostan, Laureano González-Vega, Hervé Perdry, and Éric Schost. From Newton sums to coefficients : complexity issues in characteristic p . In *MEGA’05*, 2005.
- [BJS⁺07] Tristram Bogart, Anders N Jensen, David Speyer, Bernd Sturmfels, and Rekha R Thomas. Computing tropical varieties. *Journal of Symbolic Computation*, 42(1) :54–73, 2007.
- [BL12] J. Berthomieu and R. Lebreton. Relaxed p -adic Hensel lifting for algebraic systems. In *ISSAC ’12 : Proceedings of the 2012 international symposium on Symbolic and algebraic computation*, ISSAC ’12, pages 59–66, New York, NY, USA, 2012. ACM.
- [BMMT94] Eberhard Becker, Teo Mora, Maria Grazia Marinari, and Carlo Traverso. The shape of the shape lemma. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation, ISSAC ’94, Oxford, UK, July 20-22, 1994*, pages 129–133, 1994.
- [BMSS08] Alin Bostan, François Morain, Bruno Salvy, and Éric Schost. Fast algorithms for computing isogenies between elliptic curves. *Math. Comput.*, 77(263) :1755–1778, 2008.
- [BQ12] Yves Benoist and Jean-François Quint. Introduction to random walks on homogeneous spaces. *Japanese Journal of Mathematics*, 7(2) :135–166, 2012.
- [Buc65] B. Buchberger. *Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal (An Algorithm for Finding the Basis Elements in the Residue Class Ring Modulo a Zero Dimensional Polynomial Ideal)*. PhD thesis, Mathematical Institute, University of Innsbruck, Austria, 1965. English translation in J. of Symbolic Computation, Special Issue on Logic, Mathematics, and Computer Science : Interactions. Vol. 41, Number 3-4, Pages 475–511, 2006.
- [BvdHL11] Jérémy Berthomieu, Joris van der Hoeven, and Grégoire Lecerf. Relaxed algorithms for p -adic numbers. *J. Théorie des Nombres des Bordeaux*, 23(3) :541–577, 2011.
- [Car12] Xavier Caruso. Random matrices over a DVR and LU factorization. arXiv :1212.0308, 2012.
- [CC15] A. Connes and C. Consani. Geometry of the arithmetic site. *arxiv :1502.05580*, 2015.

- [Cha13] Andrew J. Chan. Gröbner bases over fields with valuations and tropical curves by coordinate projections. *PhD Thesis, University of Warwick*, August 2013.
- [Cho83] Man-Duen Choi. Tricks or treats with the hilbert matrix. *American Mathematical Monthly*, pages 301–312, 1983.
- [CL14] Xavier Caruso and David Lubicz. Linear algebra over $\mathbb{Z}_p[[u]]$ and related rings. *LMS J. Comput. Math.*, 17(1) :302–344, 2014.
- [CLO07] David A Cox, John Little, and Donal O’Shea. *Ideals, varieties, and algorithms : an introduction to computational algebraic geometry and commutative algebra*. Springer Science & Business Media, 2007.
- [CM13] Andrew J Chan and Diane Maclagan. Gröbner bases over fields with valuations. *arxiv :1303.0729*, 2013.
- [Con15] Alain Connes. An essay on the Riemann Hypothesis. *arxiv :1509.05576*, 2015.
- [CRV14] Xavier Caruso, David Roe, and Tristan Vaccon. Tracking p -adic precision. *LMS J. Comput. Math.*, 17(suppl. A) :274–294, 2014.
- [CRV15] Xavier Caruso, David Roe, and Tristan Vaccon. p -Adic Stability In Linear Algebra. In *Proceedings of the 2015 ACM on International Symposium on Symbolic and Algebraic Computation, ISSAC 2015, Bath, United Kingdom*, pages 101–108, 2015.
- [CS95] A. Robert Calderbank and Neil JA Sloane. Modular and p -adic cyclic codes. *Designs, Codes and Cryptography*, 6(1) :21–35, 1995.
- [CS05] Aldo Conca and Jessica Sidman. Generic initial ideals of points and curves. *J. Symb. Comput.*, 40(3) :1023–1038, September 2005.
- [Del74] Pierre Deligne. La conjecture de Weil. I. *Publications Mathématiques de l’Institut des Hautes Études Scientifiques*, 43(1) :273–307, 1974.
- [Del80] Pierre Deligne. La conjecture de Weil. II. *Publications Mathématiques de l’IHÉS*, 52(1) :137–252, 1980.
- [Dix82] John D. Dixon. Exact Solution of Linear Equations using P -Adic Expansions. *Numerische Mathematik*, 40(1) :137–141, 1982.
- [DKSS13] Anindya De, Piyush P. Kurur, Chandan Saha, and Ramprasad Saptharishi. Fast integer multiplication using modular arithmetic. *SIAM J. Comput.*, 42(2) :685–699, 2013.
- [DL08] Clémence Durvye and Grégoire Lecerf. A concise proof of the Kronecker polynomial system solver from scratch. *Expositiones Mathematicae*, 26(2) :101 – 139, 2008.
- [DS04] Xavier Dahan and Éric Schost. Sharp estimates for triangular sets. In *Proceedings of the 2004 International Symposium on Symbolic and Algebraic Computation, ISSAC ’04*, pages 103–110, New York, NY, USA, 2004. ACM.
- [DSV01] Jean-Guillaume Dumas, B David Saunders, and Gilles Villard. On efficient sparse integer matrix Smith normal form computations. *Journal of Symbolic Computation*, 32(1) :71–99, 2001.
- [Dwo60] Bernard Dwork. On the rationality of the zeta function of an algebraic variety. *American Journal of Mathematics*, pages 631–648, 1960.
- [EF14] Christian Eder and Jean-Charles Faugère. A survey on signature-based Gröbner basis computations. <http://hal.inria.fr/hal-00974810>, April 2014.
- [Eis95] David Eisenbud. *Commutative Algebra : with a view toward algebraic geometry*, volume 150. Springer Science & Business Media, 1995.
- [EM07] Mohamed Elkadi and Bernard Mourrain. *Introduction à la résolution des systèmes polynomiaux*, volume 59. Springer, 2007.
- [Fau02] Jean-Charles Faugère. A new efficient algorithm for computing Gröbner bases without reduction to zero (F5). In *Proceedings of the 2002 international symposium on Symbolic and algebraic computation, ISSAC ’02*, pages 75–83, New York, NY, USA, 2002. ACM.

- [FGHR13] Jean-Charles Faugère, Pierrick Gaudry, Louise Huot, and Guénaél Renault. Polynomial Systems Solving by Fast Linear Algebra. preprint, 2013. 23 pages.
- [FGHR14] Jean-Charles Faugère, Pierrick Gaudry, Louise Huot, and Guénaél Renault. Sub-cubic Change of Ordering for Gröbner Basis : A Probabilistic Approach. In *Proceedings of the 39th International Symposium on Symbolic and Algebraic Computation*, pages 170–177, Kobe, Japon, July 2014. ACM.
- [FGLM93] Jean-Charles Faugère, Patrizia Gianni, Daniel Lazard, and Teo Mora. Efficient computation of zero-dimensional Gröbner bases by change of ordering. *Journal of Symbolic Computation*, 16(4) :329–344, 1993.
- [FL07] Jean-Charles Faugère and Ye Liang. Numerical Computation of Grobner Bases for Zero-dimensional Polynomial Ideals. In *Mathematical Aspects of Computer and Information Sciences 2007, Paris, France*, December 2007.
- [FL11a] Jean-Charles Faugère and Ye Liang. Artificial discontinuities of single-parametric Gröbner bases. *Journal of Symbolic Computation*, 46(4) :459–466, 2011.
- [FL11b] Jean-Charles Faugère and Ye Liang. Pivoting in Extended Rings for Computing Approximate Gröbner Bases. *Mathematics in Computer Science*, 5 :179–194, 2011.
- [FM11] Jean-Charles Faugère and Chenqi Mou. Fast Algorithm for Change of Ordering of Zero-dimensional Gröbner Bases with Sparse Multiplication Matrices. In *Proceedings of the 36th international symposium on Symbolic and algebraic computation*, ISSAC '11, pages 115–122, New York, NY, USA, 2011. ACM.
- [FM13] Jean-Charles Faugère and Chenqi Mou. Sparse FGLM algorithms. *CoRR*, abs/1304.1238, 2013.
- [FSEDV13] Jean-Charles Faugère, Mohab Safey El Din, and Thibaut Verron. On the complexity of Computing Gröbner Bases for Quasi-homogeneous Systems. In *Proceedings of the 38th international symposium on International symposium on symbolic and algebraic computation*, ISSAC '13, pages 189–196, New York, NY, USA, 2013. ACM.
- [Für09] Martin Fürer. Faster integer multiplication. *SIAM J. Comput.*, 39(3) :979–1005, 2009.
- [FZ02] Sergey Fomin and Andrei Zelevinsky. The Laurent phenomenon. *Advances in Applied Math.*, 28(2) :119–144, 2002.
- [GH91] Marc Giusti and Joos Heintz. Algorithmes–disons rapides–pour la décomposition d’une variété algébrique en composantes irréductibles et équidimensionnelles. In *Effective Methods in Algebraic Geometry*, pages 169–194. Springer, 1991.
- [GHW⁺06] Pierrick Gaudry, Thomas Houtmann, Annegret Weng, Christophe Ritzenthaler, and David Kohel. The 2-adic CM method for genus 2 curves with application to cryptography. In *Asiacrypt 2006*, LNCS 4284, pages 114–129. Springer, 2006.
- [Giu84] Marc Giusti. Some effectivity problems in polynomial ideal theory. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation*, EUROSAM '84, pages 159–171, London, UK, UK, 1984. Springer-Verlag.
- [GLS01] Marc Giusti, Grégoire Lecerf, and Bruno Salvy. A gröbner free alternative for polynomial system solving. *Journal of Complexity*, 17(1) :154 – 211, 2001.
- [GM89] Patrizia Gianni and Teo Mora. Algebraic solution of systems of polynomial equations using Groebner bases. In *Applied algebra, algebraic algorithms and error-correcting codes (Menorca, 1987)*, volume 356 of *Lecture Notes in Comput. Sci.*, pages 247–257. Springer, Berlin, 1989.
- [Grä93] Hans-Gert Gräbe. On lucky primes. *J. Symb. Comput.*, 15(2) :199–209, February 1993.
- [GvdHL15] Bruno Grenet, Joris van der Hoeven, and Grégoire Lecerf. Deterministic root finding over finite fields using Graeffe transforms. Manuscript (submitted), 2015.
- [Hen97] Kurt Hensel. Über eine neue Begründung der Theorie der algebraischen Zahlen. *Jahresbericht der Deutschen Mathematiker-Vereinigung*, 6 :83–88, 1897.

- [Hen66] Frederick C Hennie. On-line turing machine computations. *Electronic Computers, IEEE Transactions on*, (1) :35–44, 1966.
- [Huo13] Louise Huot. *Résolution de systèmes polynomiaux et cryptologie sur les courbes elliptiques*. PhD thesis, Université Pierre et Marie Curie (Paris VI), December 2013. <http://tel.archives-ouvertes.fr/tel-00925271>.
- [HvdHL14a] David Harvey, Joris van der Hoeven, and Grégoire Lecerf. Even faster integer multiplication. *CoRR*, abs/1407.3360, 2014.
- [HvdHL14b] David Harvey, Joris van der Hoeven, and Grégoire Lecerf. Faster polynomial multiplication over finite fields. *CoRR*, abs/1407.3361, 2014.
- [Jen] Anders N. Jensen. Gfan, a software system for Gröbner fans and tropical varieties. Available at <http://home.imf.au.dk/jensen/software/gfan/gfan.html>.
- [JLY14] A. Jensen, A. Leykin, and J. Yu. Computing tropical curves via homotopy continuation. *ArXiv e-prints*, August 2014.
- [Ked01] Kiran S. Kedlaya. Counting points on hyperelliptic curves using monsky–washnitzer cohomology. *J. Ramanujan Math. Soc.*, 16 :323–338, 2001.
- [Ked10] Kiran S. Kedlaya. *p-adic differential equations*, volume 125 of *Cambridge Studies in Advanced Mathematics*. Cambridge UP, Cambridge, UK, 2010.
- [KMRS75] E.V. Krishnamurthy, T. Mahadeva Rao, and K. Subramanian. P-adic arithmetic procedures for exact matrix computations. In *Proceedings of the Indian Academy of Sciences-Section A*, volume 82, pages 165–175. Springer, 1975.
- [Knu69] Donald E. Knuth. *The art of computer programming. Vol. 2 : Seminumerical algorithms*. Addison-Wesley Publishing Co., Reading, Mass.-London-Don Mills, Ont, 1969.
- [KO62] Anatolii Karatsuba and Yuri Ofman. Multiplication of many-digital numbers by automatic computers. *Proceedings of the USSR Academy of Sciences*, (145) :293–294, 1962.
- [Kri75] E.V. Krishnamurthy. Matrix processors using p-adic arithmetic for exact linear computations. In *Computer Arithmetic (ARITH), 1975 IEEE 3rd Symposium on*, pages 92–97. IEEE, 1975.
- [KSW04] A. Kondratyev, H.J. Stetter, and F. Winkler. Numerical Computation of Gröbner Bases. In V.G. Ghanza, E.W. Mayr, and E.V. Vorozhtov, editors, *7th Workshop on Computer Algebra in Scientific Computing, CASC-2004*, 2004.
- [KU11] Kiran S. Kedlaya and Christopher Umans. Fast polynomial factorization and modular composition. *SIAM J. Comput.*, 40(6) :1767–1802, 2011.
- [Lau04] Alan Lauder. Deformation theory and the computation of zeta functions. *Proc. London Math. Soc.*, 88(3) :565–602, 2004.
- [Laz83] Daniel Lazard. Gröbner-Bases, Gaussian Elimination and Resolution of Systems of Algebraic Equations. In *Proceedings of the European Computer Algebra Conference on Computer Algebra, EUROCAL '83*, pages 146–156, London, UK, UK, 1983. Springer-Verlag.
- [Leb13] Romain Lebreton. Relaxed Hensel lifting of triangular sets. In *MEGA'2013 (Special Issue)*, Frankfurt am Main, Germany, June 2013.
- [Lim93] Carla Limongelli. On an efficient algorithm for big rational number computations by parallel p-adics. *Journal of symbolic computation*, 15(2) :181–197, 1993.
- [LL91] Y. N. Lakshman and Daniel Lazard. On the complexity of zero-dimensional algebraic systems. In *Effective methods in algebraic geometry (Castiglione, 1990)*, volume 94 of *Progr. Math.*, pages 217–225. Birkhäuser Boston, Boston, MA, 1991.
- [LL14] Chao Lu and Xinkai Li. An introduction of multiple p-adic data type and its parallel implementation. In *Computer and Information Science (ICIS), 2014 IEEE/ACIS 13th International Conference on*, pages 303–308. IEEE, 2014.

- [LP94] Carla Limongelli and Roberto Pirastu. Exact solution of linear systems over rational numbers by parallel p -adic arithmetic. In *Parallel Processing : CONPAR 94—VAPP VI*, pages 313–323. Springer, 1994.
- [LS08] Reynald Lercier and Thomas Sirvent. On Elkies subgroups of ℓ -torsion points in elliptic curves defined over a finite field. *J. Théorie des Nombres des Bordeaux*, 20 :783–797, 2008.
- [Man06] Manfred Einsiedler, Mikhail Kapranov, and Douglas Lind. Non-archimedean amoebas and tropical varieties. *Journal für die reine und angewandte Mathematik (Crelles Journal)*, 2006(601) :139–157, 2006.
- [MM82] Ernst W Mayr and Albert R Meyer. The complexity of the word problems for commutative semigroups and polynomial ideals. *Advances in mathematics*, 46(3) :305–329, 1982.
- [Mou13] Chenqi Mou. *Solving Polynomial Systems over Finite Fields : Algorithms, Implementation and Applications*. Theses, Université Pierre et Marie Curie, May 2013.
- [MS91] Guillermo Moreno-Socias. Autour de la fonction de Hilbert-Samuel (escaliers d’idéaux polynomiaux). *Thèse, École Polytechnique*, 1991.
- [MS15] Diane Maclagan and Bernd Sturmfels. *Introduction to tropical geometry*, volume 161 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, 2015.
- [MT08] Bernard Mourrain and Philippe Trébuchet. Stable normal forms for polynomial system solving. *Theoretical Computer Science*, 409(2) :229–240, 2008.
- [Nag09] Kosaku Nagasaka. A Study on Gröbner Basis with Inexact Input. In *Proceedings of the 11th International Workshop on Computer Algebra in Scientific Computing, CASC '09*, pages 247–258, Berlin, Heidelberg, 2009. Springer-Verlag.
- [Par10] Keith Pardue. Generic sequences of polynomials. *J. Algebra*, 324(4) :579–590, 2010.
- [Pau92] Franz Pauer. On Lucky Ideals for Gröbner Basis Computations. *J. Symb. Comput.*, 14(5) :471–482, November 1992.
- [PS11] Rob Pollack and Glenn Stevens. Overconvergent modular symbols and p -adic L -functions. *Annales scientifiques de l’ENS*, 44(1) :1–42, 2011.
- [Rab60] Michael O. Rabin. Computable algebra, general theory and theory of computable fields. *Transactions of the AMS*, 95 :341–360, 1960.
- [Rob00] Alain Robert. *A course in p -adic analysis*, volume 198. Springer Science & Business Media, 2000.
- [Roc97] R. Tyrell Rockafellar. *Variational Analysis*. Grundlehren der Mathematischen Wissenschaften 317. Springer-Verlag, 1997.
- [Rou99] Fabrice Rouillier. Solving zero-dimensional systems through the rational univariate representation. *Applicable Algebra in Engineering, Communication and Computing*, 9(5) :433–461, 1999.
- [RY06] Guénaél Renault and Kazuhiro Yokoyama. A modular method for computing the splitting field of a polynomial. In *Proceedings of the 7th International Conference on Algorithmic Number Theory, ANTS’06*, pages 124–140, Berlin, Heidelberg, 2006. Springer-Verlag.
- [S⁺11] W. A. Stein et al. *Sage Mathematics Software (Version 4.7.2)*. The Sage Development Team, 2011. <http://www.sagemath.org>.
- [Sas11] Tateaki Sasaki. A Theory and an Algorithm of Approximate Gröbner Bases. In *Proceedings of the 2011 13th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC ’11*, pages 23–30, Washington, DC, USA, 2011. IEEE Computer Society.
- [Sch11] Peter Schneider. *p -Adic Lie groups*. Grundlehren der mathematischen Wissenschaften 344. Springer, Berlin, 2011.
- [Ser62] Jean-Pierre Serre. *Corps locaux*, volume 3. Hermann Paris, 1962.

- [Ser79] Jean-Pierre Serre. *Local fields*, volume 67 of *Graduate Texts in Mathematics*. Springer-Verlag, New York-Berlin, 1979. Translated from the French by Marvin Jay Greenberg.
- [SK07] Tateaki Sasaki and Fujio Kako. Computing Floating-point Gröbner Bases Stably. In *Proceedings of the 2007 International Workshop on Symbolic-numeric Computation*, SNC '07, pages 180–189, New York, NY, USA, 2007. ACM.
- [SK10] Tateaki Sasaki and Fujio Kako. Term Cancellations in Computing Floating-point Gröbner Bases. In *Proceedings of the 12th International Conference on Computer Algebra in Scientific Computing*, CASC'10, pages 220–231, Berlin, Heidelberg, 2010. Springer-Verlag.
- [Som89] Michael Somos. Problem 1470. *Cruz Mathematicorum*, 15 :208, 1989.
- [SS98] Kiyoshi Shirayanagi and Moss Sweedler. Remarks on automatic algorithm stabilization. *J. Symb. Comput.*, 26(6) :761–765, December 1998.
- [Ste] H.J. Stetter. Approximate Gröbner bases – an impossible concept? In *Proceedings of SNC2005 (Symbolic-Numeric Computation)*, Xi'an, China, SNC, pages 235–236.
- [Sva14] Jules Svartz. *Résolution de systèmes polynomiaux structurés de dimension zéro*. PhD thesis, Université Pierre et Marie Curie (Paris VI), Octobre 2014. <https://tel.archives-ouvertes.fr/tel-01147484>.
- [Tod54] John Todd. The condition of the finite segments of the hilbert matrix. *Contributions to the solution of systems of linear equations and the determination of eigenvalues*, 39 :109–116, 1954.
- [TW95] Richard Taylor and Andrew Wiles. Ring-theoretic properties of certain Hecke algebras. *Ann. of Math. (2)*, 141(3) :553–572, 1995.
- [TZ02] Carlo Traverso and Alberto Zanon. Numerical stability and stabilization of Gröbner basis computation. In *Proceedings of the 2002 International Symposium on Symbolic and Algebraic Computation*, ISSAC '02, pages 262–269, New York, NY, USA, 2002. ACM.
- [Vac14] Tristan Vaccon. Matrix-F5 algorithms over finite-precision complete discrete valuation fields. In *Proceedings of the 2014 ACM on International Symposium on Symbolic and Algebraic Computation*, ISSAC '14, Kobe, Japan, pages 397–404, 2014.
- [Vac15] Tristan Vaccon. Matrix-F5 Algorithms and Tropical Gröbner Bases Computation. In *Proceedings of the 2015 ACM on International Symposium on Symbolic and Algebraic Computation*, ISSAC 2015, Bath, United Kingdom, pages 355–362, 2015.
- [vdH97] Joris van der Hoeven. Lazy multiplication of formal power series. In *Proceedings of the 1997 international symposium on Symbolic and algebraic computation*, pages 17–20. ACM, 1997.
- [vdH02] Joris van der Hoeven. Relax, but don't be too lazy. *J. Symbolic Comput.*, 34(6) :479–542, 2002.
- [vdH07] Joris van der Hoeven. New algorithms for relaxed multiplication. *J. Symbolic Comput.*, 42(8) :792–802, 2007.
- [vdHLM⁺12] Joris van der Hoeven, Grégoire Lecerf, Bernard Mourrain, Philippe Trébuchet, Jérémy Berthomieu, Daouda Niang Diatta, and Angelos Mantzaflaris. Mathemagix : The quest of modularity and efficiency for symbolic and certified numeric computation? *ACM Commun. Comput. Algebra*, 45(3/4) :186–188, January 2012.
- [VH02] Mark Van Hoeij. Factoring polynomials and the knapsack problem. *Journal of Number theory*, 95(2) :167–189, 2002.
- [VZGG13] Joachim Von Zur Gathen and Jürgen Gerhard. *Modern computer algebra*. Cambridge university press, 2013.
- [Win88] Franz Winkler. A p-adic approach to the computation of Gröbner bases. *J. Symb. Comput.*, 6(2-3) :287–304, December 1988.
- [Zas69] Hans Zassenhaus. On Hensel factorization, I. *Journal of Number Theory*, 1(3) :291–311, 1969.

"If you have enough book space, I
don't want to talk to you"

Terry Pratchett, unsourced

"Everything not saved will be lost."

Nintendo "Quit Screen" message,
cited in *The End Games*, by
T.Michael Martin

"Thank you for playing"

Nintendo